JEAN-FRANCOIS
SEHAN

ETUDES POUR ZX 81

20 PROGRAMMES EN BASIC

EDITIONS DU P.S.I.

PROGRAMMES

PROGRAMMES

ETUDES POUR
ZX 81

COLLECTION LANGAGES

Langages de programmation - Stéphane Berche et Claude Lhermitte
Le langage ADA - Daniel-Jean David
Programmer en APL - Daniel-Jean David
Programmer en Assembleur - Alain Pinaud
Programmer en BASIC - Michel Plouin
Le Basic et ses fichiers - tomes 1 et 2 - Jacques Boisgontier
Programmer en FORTRAN - Daniel-Jean David
Programmer en L.S.E. - Stéphane Berche et Yves Noyelle
Programmer en PASCAL - Daniel-Jean David et Jean-Luc Deschamps
Comment programmer - Jean-Claude Barbance
Comprendre les microprocesseurs - Roland Dubois

COLLECTION MATERIELS

La découverte de l'Applesoft - tome 1 - Frédéric Lévy et Dominique Schraen
La découverte de l'Applesoft - tome 2 - frédéric Lévy
La pratique de l'Apple II - volumes I et II - Nicole Bréaud-Pouliquen
La pratique de l'Apple II - volume III - Nicole Bréaud-Pouliquen et Daniel-Jean David
La découverte du Goupil - Jean-Yves Michel
La pratique du MZ-80 A et K - Jean-Pierre Lhoir
La découverte du PC-1211 - Jean-Pierre Richard
La découverte du PET/CBM - Daniel-Jean David
La pratique du PET/CBM - volumes I et II - Daniel-Jean David
La pratique du TRS-80 - volumes I, II et III - Pierre Giraud et Alain Pinaud
La découverte de la TI-57 - Xavier de La Tullaye
La découverte du VIC - Daniel-Jean David
La pratique du ZX81 - Xavier Linant de Bellefond

COLLECTION GUIDES PRATIQUES

L'APL sur TRS-80 - Claude Nowakowski
CP/M pas a pas - Alain Pinaud
LISP sur Apple II - Nicole Bréaud-Pouliquen
La réalisation des programmes - Michel Benelfoul
Méthodes de calcul numérique - Claude Nowakowski

COLLECTION PROGRAMMES

Etudes pour ZX81 - Jean-François Séhan
Jeux, trucs et comptes pour PET/CBM - Michel Benelfoul
Mathématiques et statistiques - Hervé Haut
Modèles pratiques de décision - tomes 1 et 2 - Jean-Pierre Blanger
Récréations pour TI-57 - tomes 1 et 2 - Jacques Deconchat
Variations pour PC-1211 - Jean-François Séhan

COLLECTION MEMENTOS

Clefs pour le PET/CBM - Daniel-Jean David

TRADUCTIONS

Les graphiques sur TRS-80 - Don Inman traduit par Alain Pinaud
Le petit livre du ZX81 - Trévor Toms traduit par Ghislaine Lapeyre

INITIATION

Visa pour l'informatique - Jean-Michel Jégo
Mon ordinateur - Jean-Claude Barbance

ISBN : 2-86595-42-5

Imprimé en France

JEAN-FRANÇOIS
SEHAN

ETUDES POUR ZX 81

Editions du P.S.I.
41-45, rue Jacquard
77400 Lagny/Marne

Jean-François SEHAN *est gérant d'Info-bat, société de service et de conseils en informatique, spécialisée dans la gestion sur micro-ordinateur des petites et moyennes entreprises. Il coordonne, en tant que directeur technique, les études des matériels et logiciels des systèmes clé en main proposés par sa société.*

S O M M A I R E

PRESENTATION

Le ZX81 ouvre aujourd'hui une nouvelle ère dans le domaine de la micro-informatique. Pourtant, d'un prix modique, celui-ci apporte de nombreuses possibilités d'utilisation qu'il s'agissent de jeux, de gestion ou tout simplement de programmes pratiques.

C'est pour en faire la démonstration que l'auteur a choisi pour ce livre des domaines d'application très diversifiés. La priorité est toutefois donnée aux jeux, liant ainsi l'utile à l'agréable.

Cependant, ce livre n'est pas seulement une liste exhaustive de programmes mais plutôt un ensemble d' "Etudes" de cas pour lesquels vous trouverez donc dans chaque chapitre une présentation et des exemples de ce que l'on obtient sur l'écran.

La structure générale des programmes est présentée sous forme d'un ou plusieurs organigrammes détaillés, parfois accompagné d'un organigramme général pour une meilleure vue d'ensemble.

Chaque ligne est ensuite abondamment commentée, expliquant les astuces utilisées. Le tout est associé à un tableau d'identification de variables ce qui permet une recherche ultérieure plus facile.

Toutes ces explications sont données dans un but essentiellement pédagogique pour construire vous-mêmes vos applications personnelles.

Mais sans plus attendre, passons aux "choses sérieuses".

Avertissement

Vue la complexité des programmes de ce livre, il est nécessaire de posséder l'extension 16 ko du ZX81, la version de base n'ayant qu'un petit ko de mémoire vive.

Seul le programme "MACRO-LETTRES" nécessite une imprimante. Tous les autres programmes peuvent s'en passer. Mais, si vous en avez une, il sera particulièrement intéressant de l'utiliser pour des copies d'écran dans les programmes BIORYTHME, CALENDRIER, GRAPHIQUES, GESTION DE FICHIER, TRICOT et FICHIER TELEPHONE.

Dès que les programmes sont entrés dans le ZX81, sauvez-les sur votre magnétophone. Pour les programmes très longs, il est conseillé d'effectuer une première sauvegarde au milieu de la frappe du programme. N'attendez pas qu'il soit entièrement entré et testé... Une coupure de courant est si vite arrivée !

Méfiez-vous ! L'instruction ***FAST*** *coupe complètement le signal vidéo de votre téléviseur, laissant, pendant quelques instants, l'impression que votre ZX81 ne marche plus. Il est toujours possible d'arrêter le programme en cours par la touche* ***BREAK*** *si vous pensez qu'il est arrivé une catastrophe et avant d'effectuer une manoeuvre qui perdrait définitivement votre programme.*

Pour recopier les lignes comportant des caractères graphiques dont le découpage ne vous paraîtrait pas clair, reportez-vous aux exemples donnés avec chaque listing. Aidez-vous aussi des lignes précédentes pour repérer chaque caractère, en comparant avec le nombre et la position des caractères de ces autres lignes.

Jack-Pot

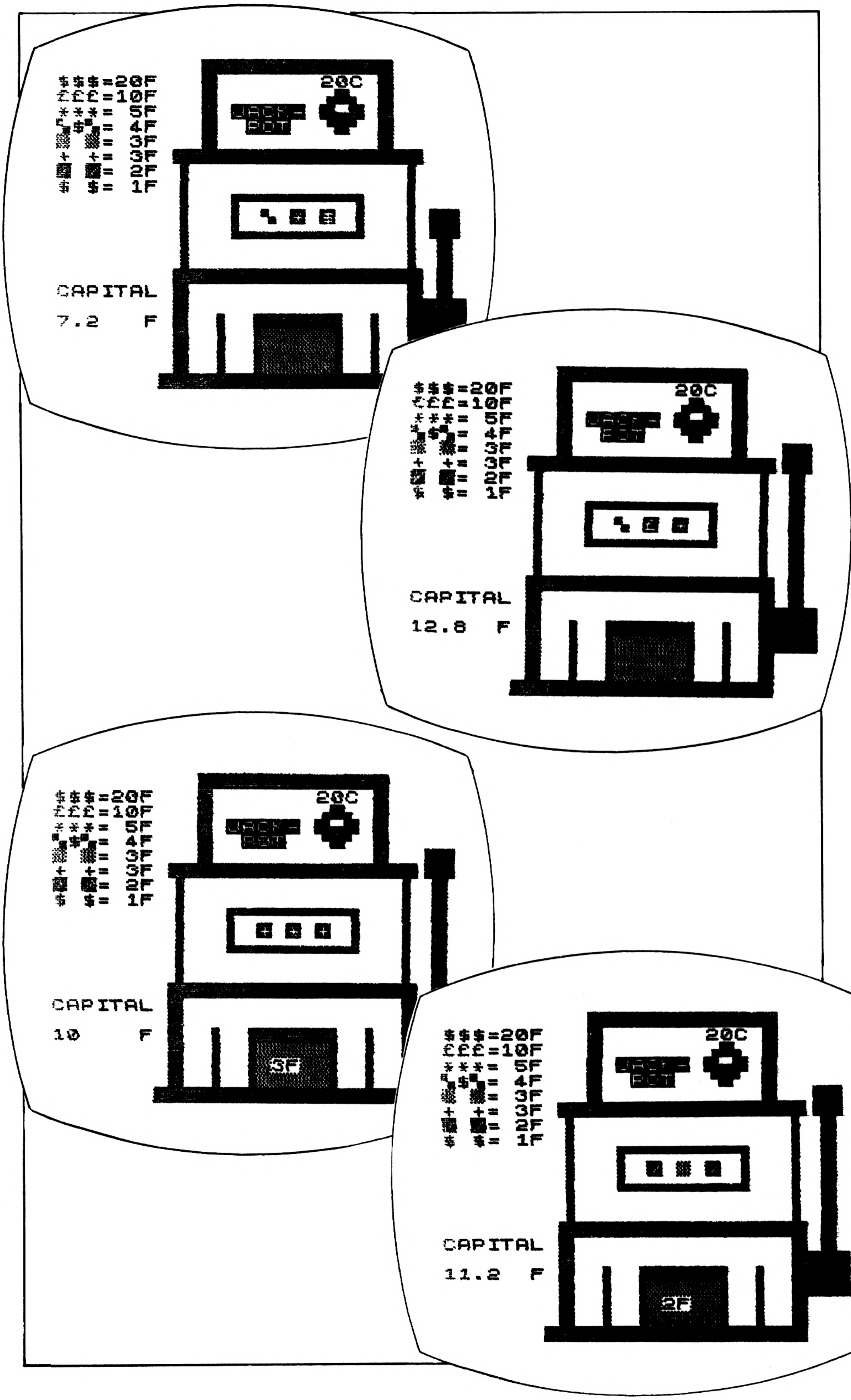
Voilà un jeu qui peut paraître tout d'abord passif puisque le joueur intervient très peu ; mais attention pour le joueur acharné : il risque d'en perdre le sommeil !

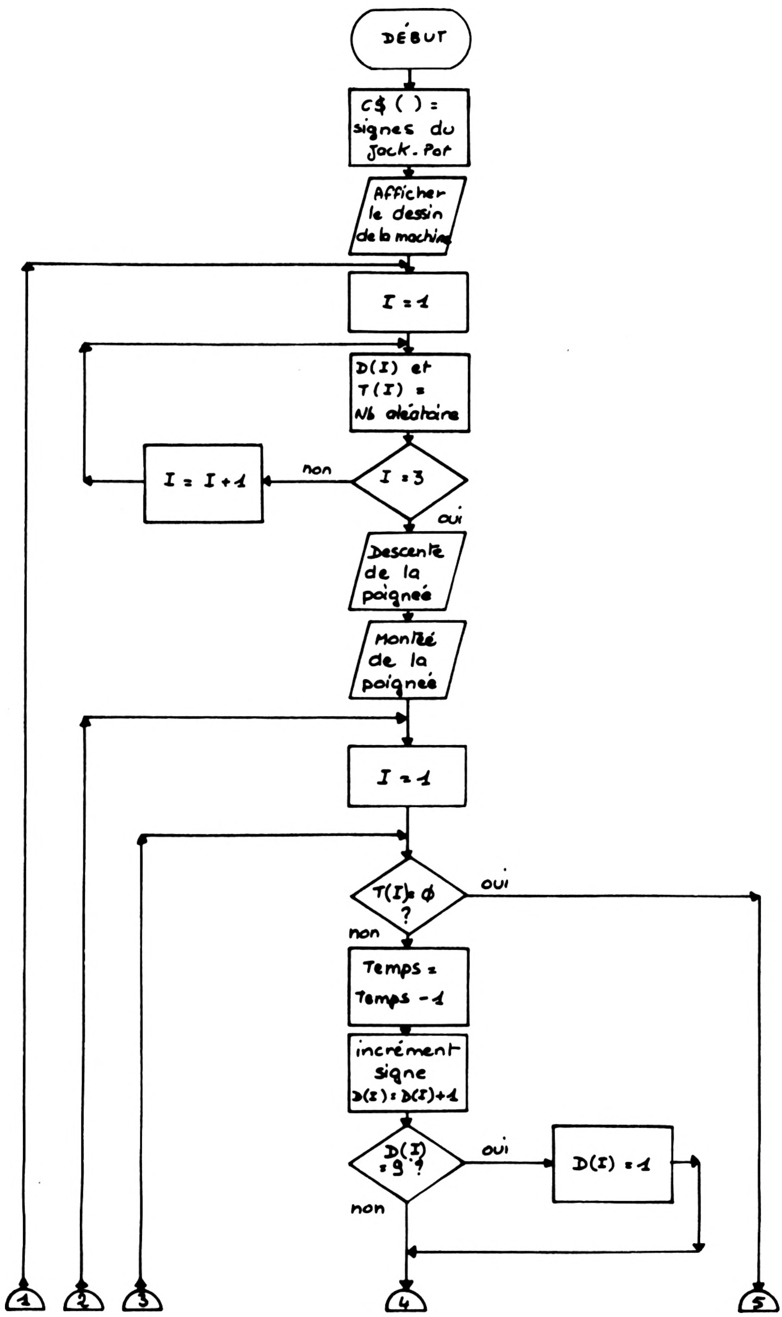
Après avoir tapé **RUN**, le JACK-POT se dessine, puis les combinaisons gagnantes, votre capital initial de 10,00 F., et c'est parti : la manette s'abaisse, les roulettes tournent.

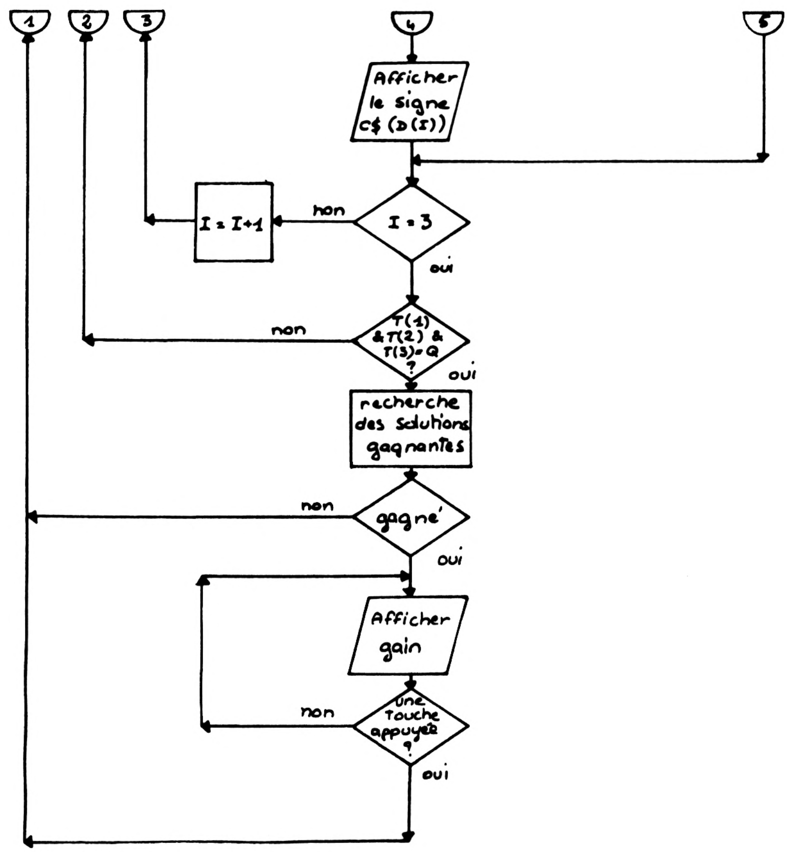
Quand elles s'immobilisent, si vous n'avez rien gagné, votre capital se décrémente de 0,20 F. Il suffit d'appuyer sur n'importe quelle touche pour relancer le jeu.

Quand une combinaison gagnante apparaît à l'arrêt des roulettes, votre gain s'affiche en clignotant en bas du JACK-POT et votre capital augmente.

Patience, patience, ce JACK-POT, digne de ses ancêtres, peut vous prendre beaucoup de temps...


$$$=20F
£££=10F
***= 5F
20C
CAPITAL
7.2 F
$$$=20F
£££=10F
***= 5F
20C
CAPITAL
12.8 F
$$$=20F
£££=10F
***= 5F
20C
CAPITAL
10 F
3F
$$$=20F
£££=10F
***= 5F
20C
CAPITAL
11.2 F
2F


DÉBUT
C$ ( ) = signes du Jack. Pot
Afficher le dessin de la machine
I = 1
D(I) et T(I) = Nb aléatoire
I = 3
non
oui
I = I + 1
Descente de la poignée
Montée de la poignée
I = 1
T(I) = Ø ?
oui
non
Temps = Temps - 1
incrément signe D(I) = D(I)+1
D(I) = 9 ?
oui
non
D(I) = 1
1
2
3
4
5


1
2
3
4
5
Afficher le signe C$ (D(I))
I = 3
non
I = I+1
oui
T(1) & T(2) & T(3) = Q ?
non
oui
recherche des solutions gagnantes
gagné
non
oui
Afficher gain
une Touche appuyée ?
non
oui

```
  1 REM JACK POT
  2 REM ----------------------
  5 DIM C$(8,1)
  6 DIM T(3)
  7 DIM D(3)
  8 LET C=10
 10 LET C$(1)="█"
 11 LET C$(2)="█"
 12 LET C$(3)="█"
 13 LET C$(4)="█"
 14 LET C$(5)="█"
 15 LET C$(6)="█"
 16 LET C$(7)="█"
 17 LET C$(8)="█"
 18 REM ----------------------
 19 REM DESSIN DE LA MACHINE
 20 PRINT TAB (10);
 30 FOR I=1 TO 13
 40 PRINT "█";
 50 NEXT I
 60 FOR I=1 TO 17
 70 PRINT AT 6,7+I;"█";
 80 PRINT AT 14,7+I;"█";
 90 PRINT AT 21,6+I;"█";
100 NEXT I
110 PRINT AT 21,23;"█";
120 PRINT AT 21,24;"█"
130 GOSUB 900
200 FOR I=1 TO 7
210 PRINT AT 6+I,8;" █"
220 PRINT AT 6+I,24;"█ "
230 PRINT AT 14+I,8;"█"
240 PRINT AT 14+I,24;"█"
250 PRINT AT -1+I,10;"█"
260 PRINT AT -1+I,22;"█"
270 NEXT I
300 PRINT AT 2,18;" ██ "
310 PRINT AT 3,18;"█  █"
320 PRINT AT 4,18;" ██ "
330 PRINT AT 3,12;"██████"
340 PRINT AT 4,13;"████"
350 PRINT AT 1,18;"20C"
400 FOR I=1 TO 4
410 PRINT AT 16+I,11;"█  ██████████
█"
420 REM PRINT AT 16+I,19;"█ "
425 REM PRINT AT 16+I,14;"██████

430 NEXT I
440 PRINT AT 17,14;"████████"
450 PRINT AT 1,0;"$$$=20F"
460 PRINT AT 2,0;"£££=10F"
470 PRINT AT 3,0;"***= 5F"
480 PRINT AT 4,0;"█$█= 4F"
490 PRINT AT 5,0;"█ █= 3F"
500 PRINT AT 6,0;"+ += 3F"
510 PRINT AT 7,0;"█ █= 2F"
520 PRINT AT 8,0;"$ $= 1F"
600 PRINT AT 9,12;"█████████████"
610 PRINT AT 10,12;"█            █"
620 PRINT AT 11,12;"██████████████"
630 PRINT AT 16,25;"█████"
640 PRINT AT 17,25;"█████"
650 PRINT AT 18,25;"█████"
655 PRINT AT 15,0;"CAPITAL"
656 PRINT AT 17,1;C;"    F"
```

```
 659 REM ---------------------
 660 REM JEU
 661 FOR I=1 TO 3
 662 LET D(I)=INT (RND*8)+1
 663 LET T(I)=INT ((RND*20)+5)
 664 NEXT I
 665 GOSUB 800
 666 GOSUB 900
 700 FOR I=1 TO 3
 703 IF T(I)=0 THEN GOTO 770
 705 LET T(I)=T(I)-1
 715 LET D(I)=D(I)+1
 720 IF D(I)=9 THEN LET D(I)=1
 730 PRINT AT 10,12+(I*2);C$(D(I))
 770 NEXT I
 780 IF T(1)=0 AND T(2)=0 AND T(3)=0 THEN GOTO 785
 781 GOTO 700
 785 GOSUB 1000
 790 GOTO 661
 798 REM ---------------------
 799 REM DESCENTE DE LA POIGNEE
 800 FOR I=5 TO 13
 810 PRINT AT I,25;"   "
 820 PRINT AT I+1,25;" █ "
 830 PRINT AT I+2,25;" █ "
 840 NEXT I
 850 RETURN
 880 REM ---------------------
 890 REM MONTEE DE LA POIGNEE
 900 FOR I=13 TO 5 STEP -1
 910 PRINT AT I,25;"▄▄ "
 920 PRINT AT I+1,25;" ▀█ "
 930 PRINT AT I+2,25;" █ "
 940 NEXT I
 950 RETURN
 980 REM ---------------------
 990 REM TOTAL DES GAINS
1000 LET C=C-.20
1005 LET CAP=C
1010 LET D=D(1)*100+D(2)*10+D(3)
1020 LET D1=D(1)*100+D(3)
1030 IF D=333 THEN LET C=C+20
1040 IF D=666 THEN LET C=C+10
1050 IF D=222 THEN LET C=C+5
1060 IF D=535 THEN LET C=C+4
1070 IF D1=707 THEN LET C=C+3
1080 IF D1=101 THEN LET C=C+3
1090 IF D1=808 THEN LET C=C+2
1100 IF D1=303 THEN LET C=C+1
1110 PRINT AT 17,0;C;"  "
1120 IF C>CAP THEN GOTO 1150
1130 IF INKEY$="" THEN GOTO 1130
1140 RETURN
1148 REM -------------------
1149 REM AFFICHAGE DES GAINS
1150 LET CAP=C-CAP
1160 PRINT AT 19,15;"██████"
1170 IF INKEY$<>"" THEN RETURN
1180 PRINT AT 19,15;CAP;"F"
1190 GOTO 1160
```

LE PROGRAMME

Lignes 5 à 17

On déclare un tableau C$ (8,1) contenant les symboles du JACK-POT.

Lignes 19 à 656

Le dessin de la machine est entièrement élaboré entre ces lignes, excepté la poignée qui est déterminée dans le sous-programme en 880.

Lignes 659 à 790

Le jeu entier est défini dans cette partie. Deux tableaux y sont établis :

- D() donne le symbole de départ à afficher. Les symboles suivants apparaissent dans l'ordre donné aux lignes 5 à 17.
- T() donne le temps, c'est-à-dire le nombre de symboles qui vont défiler sur la machine.

Dès que ces deux tableaux sont définis, on manoeuvre la poignée en utilisant les sous-programmes en 800 et 900 (respectivement descente et montée), puis on affiche les 3 symboles à partir de D() pour un temps T(), donc (C$(D(I))).

Lignes 1000 à 1140

Total des gains :

- On commence par décrémenter le capital C ; puis, on place dans CAP le capital restant.
- En additionnant D(1) à D(3), on obtient un chiffre qui sert de comparaison avec les solutions gagnantes du JACK-POT (variable D). La variable D1 ne prend en considération que les deux extrémités d'une combinaison qui ne compte que deux symboles gagnants.
- On compare D et D1 aux solutions gagnantes et on incrémente le capital (variable C). Quand C est supérieur à CAP, on gagne quelque chose.
- On va alors à la ligne 1150 pour afficher le montant des gains ; sinon, on retourne à un nouveau jeu.

Lignes 1149 à 1190

On affiche successivement le gain et un rectangle gris, ce qui donne un effet de clignotement ; ceci tant que l'on n'appuie pas sur une touche.

DATE	NOM DU PROGRAMME : JACK-POT	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	Programme	Cassette	Disquette	LONG.	OBSERVATIONS
C	COMPTEUR		X				Capital
C$(,)	TABLEAU		X				Contient les signes du Jack.Pot
CAP	VARIABLE		X				Capital avant totalisation des
							gains
D	VARIABLE		X				Somme des combinaisons
							sorties
D(,)	TABLEAU		X				Signe à afficher
I	INDICE		X				Boucle FOR / NEXT
T(,)	TABLEAU		X				Temps de défilement des
							signes.

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

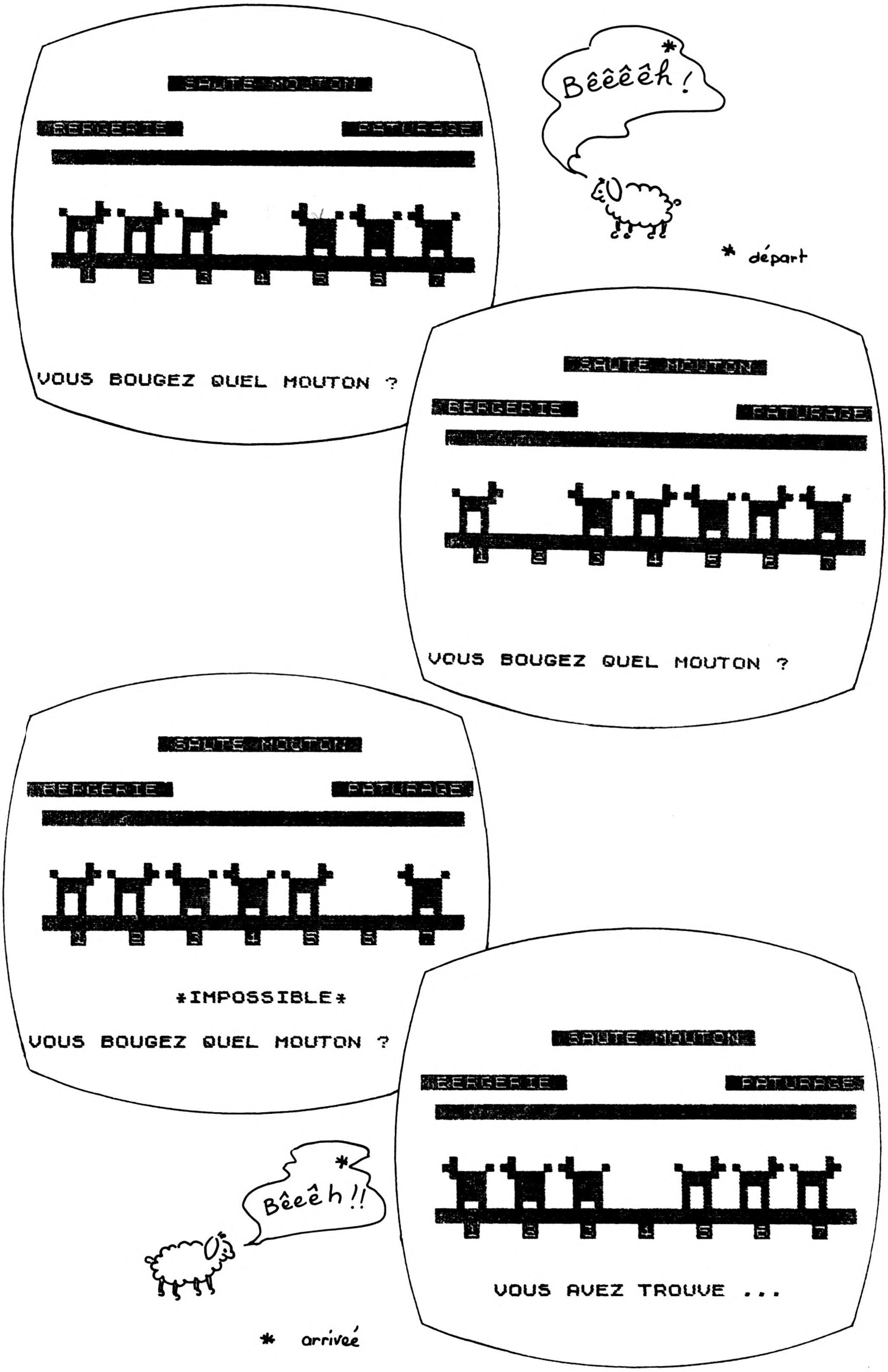
Saute mouton

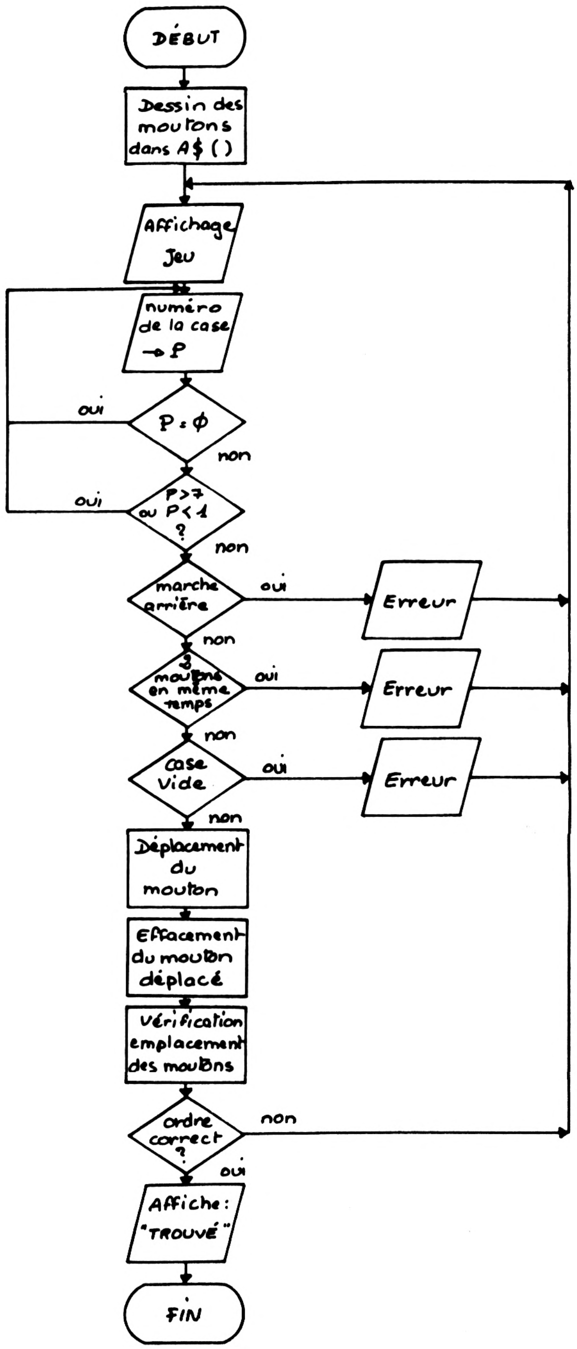
Voilà en face de vous, à gauche, trois moutons maigres dans leur bergerie et, à droite, trois moutons gras dans le paturage.

Comment faire revenir les moutons gras dans la bergerie et faire arriver les moutons maigres au paturage ?

Les moutons ne peuvent sauter qu'un seul autre mouton à la fois et ne peuvent reculer.

Vous avez essayé et vous êtes bloqué ? Alors tapez sur zéro et recommencez !


SAUTE MOUTON
BERGERIE
PATURAGE
1 2 3 4 5 6 7
VOUS BOUGEZ QUEL MOUTON ?
Bêêêh !
* départ
SAUTE MOUTON
BERGERIE
PATURAGE
1 2 3 4 5 6 7
VOUS BOUGEZ QUEL MOUTON ?
SAUTE MOUTON
BERGERIE
PATURAGE
1 2 3 4 5 6 7
IMPOSSIBLE
VOUS BOUGEZ QUEL MOUTON ?
SAUTE MOUTON
BERGERIE
PATURAGE
1 2 3 4 5 6 7
VOUS AVEZ TROUVE ...
Bêêh !!
* arrivée


DÉBUT
Dessin des moutons dans A$ ( )
Affichage Jeu
numéro de la case → P
oui
P = Ø
non
oui
P>7 ou P<1 ?
non
marche arrière
oui
Erreur
non
3 moutons en même temps
oui
Erreur
non
case vide
oui
Erreur
non
Déplacement du mouton
Effacement du mouton déplacé
Vérification emplacement des moutons
ordre correct ?
non
oui
Affiche: "TROUVÉ"
FIN

```
  10 REM SAUTE MOUTON
  20 DIM K(7)
  30 DIM A$(3,4,4)
  40 PRINT
  50 PRINT TAB 10;" SAUTE MOUTON 
 "
  70 PRINT
  80 PRINT
  90 PRINT "  BERGERIE  ";TAB 22;
" PATURAGE "
 100 LET A$(1,1)="    "
 110 LET A$(1,2)="    "
 120 LET A$(1,3)="    "
 130 LET A$(1,4)="    "
 200 LET A$(2,1)="    "
 210 LET A$(2,2)="    "
 220 LET A$(2,3)="    "
 230 LET A$(2,4)="    "
 240 LET L=4
 245 REM -----------------------
 246 REM AFFICHAGE JEU
 250 FOR I=2 TO 30
 260 PRINT AT 13,I;"█"
 270 PRINT AT 6,I;"█"
 280 NEXT I
 290 FOR X=1 TO 7
 300 PRINT AT 14,X*4;CHR$ (X+156
)
 310 LET C=1
 320 IF X=4 THEN LET C=3
 330 IF X>4 THEN LET C=2
 340 GOSUB 1000
 350 LET K(X)=C
 360 NEXT X
 370 REM -----------------------
 375 REM JEU
 380 PRINT AT 21,0;" VOUS BOUGEZ
QUEL MOUTON ?"
 390 INPUT P
 400 IF P=0 THEN GOTO 740
 410 IF P>7 OR P<1 THEN GOTO 380
 420 IF L-2>P OR L+2<P THEN GOTO
800
 430 IF K(P)=1 AND L<P THEN GOTO
800
 440 IF K(P)=2 AND L>P THEN GOTO
800
 450 IF K(P)=3 THEN GOTO 800
 460 IF K(L)<>3 THEN GOTO 800
 470 LET K(L)=K(P)
 480 LET K(P)=3
 490 LET C=3
 500 LET X=P
 510 GOSUB 1000
 520 LET C=K(L)
 530 LET X=L
 540 GOSUB 1000
 550 LET L=P
 560 IF K(1)=2 AND K(2)=2 AND K(
3)=2 AND K(4)=3 THEN GOTO 700
 570 GOTO 380
 600 REM -----------------------
 610 REM FIN
 700 FOR I=1 TO 150
 710 PRINT AT 18,6;"VOUS AVEZ TR
OUVE ..."
 720 PRINT AT 18,6;"
 "
 730 NEXT I
 740 CLS
 750 RUN
 780 REM -----------------------
 790 REM MESSAGE ERREUR
 800 FOR I=1 TO 40
 810 PRINT AT 18,11;"*IMPOSSIBLE
*"
 820 PRINT AT 18,11;"
"
 830 NEXT I
 840 GOTO 380
 980 REM -----------------------
 990 REM DEPLACEMENT MOUTON
1000 FOR Y=1 TO 4
1010 PRINT AT Y+8,2+(X-1)*4;A$(C
,Y)
1020 NEXT Y
1030 RETURN
```

LE PROGRAMME

Lignes 20 à 240

Après déclaration du tableau A$, on y place les moutons maigres et gras.

La variable L est égale à 4 en ligne 240, ce qui correspond au numéro de la case libre au départ.

Lignes 250 à 360

Ces lignes dessinent, sur l'écran, le jeu tel qu'il apparaît dans les exemples. On utilise pour cela le sous-programme à la ligne 1000 qui dessine les moutons.

Lignes 380 à 570

On entre dans la variable P le numéro du mouton à déplacer. Si ce nombre est erroné (égal à 0, inférieur à 1 ou supérieur à 7), il n'est pas accepté, mais le programme n'envoie pas de message d'erreur.

Par contre, si la case libre (variable) est éloignée de plus de 2 cases, les lignes 800 à 840 affichent un message d'erreur ; de même, si le mouton à déplacer veut reculer.

Quand la variable P est testée, on utilise le sous-programme en ligne 1000 pour déplacer le mouton sur l'écran.

On teste si les moutons sont bien placés avant d'aller au programme de fin de jeu.

Dans le cas contraire, on boucle sur la ligne 380 pour un nouveau déplacement de mouton.

Lignes 700 à 750

Dans le cas d'une erreur de déplacement, on affiche le message "IMPOSSIBLE" 40 fois avant de retourner en ligne 380.

SOUS-PROGRAMME

Lignes 1000 à 1030

Ce sous-programme dessine le mouton pointé par la variable C à la position pointée par X.

Rappelons que C=1 correspond aux moutons maigres, C=2 aux moutons gras et C=3 à une case vide.

DATE	NOM DU PROGRAMME : SAUTE - MOUTON	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A$( )	TABLEAU		X				Représentation graphique des
							moutons
C	VARIABLE		X				type de mouton C = 1 mouton maigre
							C = 2 mouton gras C = 3 case vide
I	INDICE		X				Boucle FOR / NEXT
K( )	TABLEAU		X				Emplacement des moutons
L	VARIABLE		X				Numéro de la case libre
P	VARIABLE	X					Numéro du mouton à déplacer
X	VARIABLE		X				{ position du mouton à
Y	VARIABLE		X				dessiner

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Awari

AWARI est un combat stratégique entre l'ordinateur et vous.

Le ZX81 vous demande d'entrer votre prénom ; puis, il dessine le jeu sur l'écran de cette façon :

EXEMPLE 1 :

```
__|_3_|_3_|_3_|_3_|_3_|_3_|__
 0|       PARTIE  1       |0
__|_______________________|__
  | 3 | 3 | 3 | 3 | 3 | 3 |
```

Tout au cours d'une partie, l'ordinateur affichera votre prénom, puis un message, soit :

PRENOM
JE JOUE EN

ou :

PRENOM
QUEL EST VOTRE MOUVEMENT ?

L'ordinateur joue sur les cases du haut et le joueur sur les cases du bas.

Il y a trois pions au départ dans chaque case. Le jeu consiste à déplacer les trois pions d'une case de un en un dans les cases suivantes et dans le sens inverse des aiguilles d'une montre :

EXEMPLE 2 :
MOUVEMENT

```
__|_3_|_3_|_3_|_3_|_3_|_3_|__
 0|       PARTIE  1       | 0
  |                       |+1
__|_______________________|__
  | 3 | 3 | 3 | 3 | 3 | 3 | ↗
               -3   +1  +1
               --->  --->
```

EXEMPLE 3 :
RESULTAT

```
   __|_3_|_3_|_3_|_3_|_3_|_3_|__
    0|       PARTIE  1        |1
   __|________________________|__
     | 3 | 3 | 3 | 0 | 4 | 4  |
```

Il conviendra de considérer les cases sous les numéros suivants :

EXEMPLE 4 :

```
                          sens du jeu
                       <------------
                   6   5   4   3   2   1
                  --- --- --- --- --- ---
                __|_3_|_3_|_3_|_3_|_3_|_3_|__
         total                              total
          de 0                             0 du
 l'ordinateur__|_______________________|__joueur
               | 3 | 3 | 3 | 3 | 3 | 3 |
                  --- --- --- --- --- ---
                   1   2   3   4   5   6
                       ------------>
                          sens du jeu
```

Ainsi, dans l'exemple 2, la case mouvementée était la case 4 :

PRENOM
QUEL EST VOTRE MOUVEMENT ?

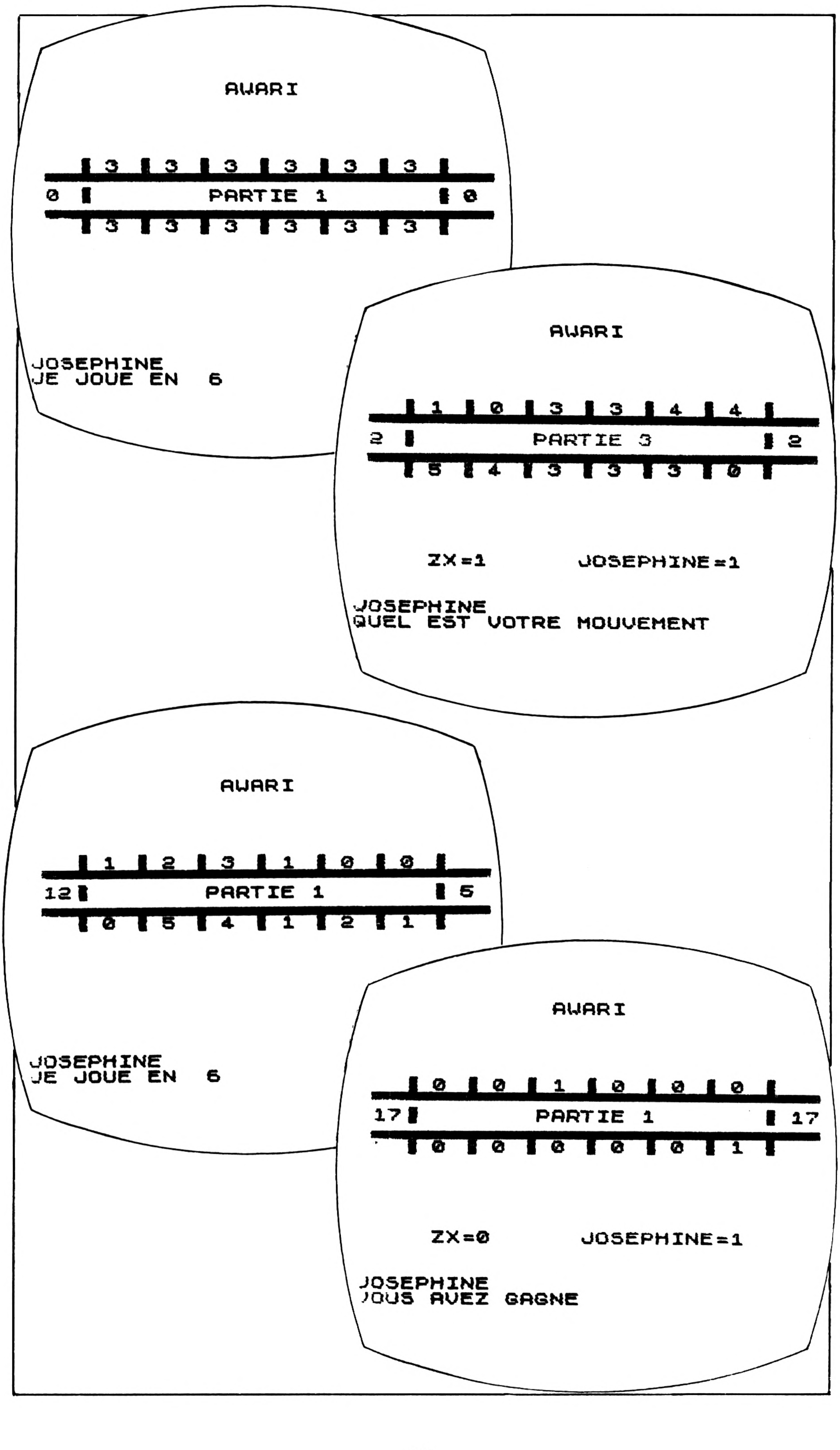
Taper 4.

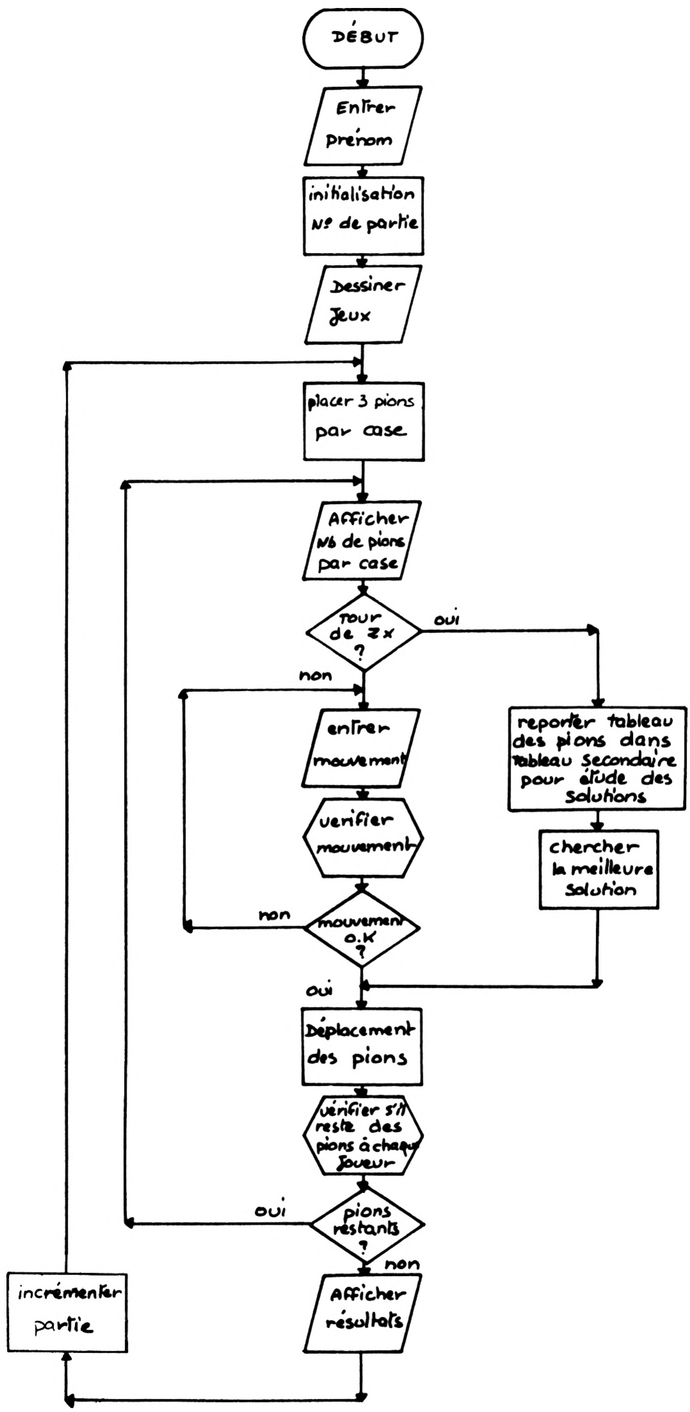
Si le dernier pion placé s'arrête dans une case TOTAL, le joueur doit jouer. Si le dernier pion s'arrête avant ou après cette case, c'est au joueur adverse de jouer. Dans l'exemple 2, c'est au joueur (et non à l'ordinateur) de rejouer.

La partie s'arrête quand toutes les cases de l'un des deux joueurs sont vides. Il y a alors affichage du nombre de parties gagnées par joueur.

Pour une autre partie, le jeu initial se redessine et le compteur de parties s'incrémente.

A vous de jouer !


AWARI
3 3 3 3 3 3
0 PARTIE 1 0
3 3 3 3 3 3
JOSEPHINE
JE JOUE EN 6
AWARI
1 0 3 3 4 4
2 PARTIE 3 2
5 4 3 3 3 0
ZX=1 JOSEPHINE=1
JOSEPHINE
QUEL EST VOTRE MOUVEMENT
AWARI
1 2 3 1 0 0
12 PARTIE 1 5
0 5 4 1 2 1
JOSEPHINE
JE JOUE EN 6
AWARI
0 0 1 0 0 0
17 PARTIE 1 17
0 0 0 0 0 1
ZX=0 JOSEPHINE=1
JOSEPHINE
VOUS AVEZ GAGNE


DÉBUT
Entrer prénom
initialisation N° de partie
Dessiner Jeux
placer 3 pions par case
Afficher Nb de pions par case
Tour de ZX ?
oui
non
reporter tableau des pions dans tableau secondaire pour étude des solutions
chercher la meilleure solution
entrer mouvement
vérifier mouvement
mouvement O.K ?
non
oui
Déplacement des pions
vérifier s'il reste des pions à chaque joueur
pions restants ?
oui
non
Afficher résultats
incrémenter partie

```
  10 REM AWARI
  50 DIM B(14)
  60 DIM K(14)
  70 DIM G(14)
  80 DIM H(14)
  90 PRINT "VOTRE PRENOM"
 100 INPUT B$
 110 LET PA=1
 120 LET T1=0
 130 LET T2=0
 140 LET E=INT (RND+.5)
 150 CLS
 160 PRINT AT 1,13;"AWARI"
 170 FOR I=1 TO 13
 180 LET B(I)=3
 190 NEXT I
 200 LET B(7)=0
 208 PRINT AT 8,2;" ";TAB 30;" "
 210 LET B(14)=0
 212 FOR I=1 TO 30
 214 PRINT AT 7,I;"▀";AT 9,I;"▄"
 216 NEXT I
 218 PRINT AT 19,0;B$
 220 PRINT AT 8,1;B(14);TAB 3;"▌
        PARTIE ";PA;TAB 27;"▌ ";
B(7)
 230 FOR J=13 TO 8 STEP -1
 240 PRINT AT 6,3+((13-J)*4);"▌
";B(J);
 250 NEXT J
 260 PRINT AT 6,27;"▌"
 270 FOR J=1 TO 6
 280 PRINT AT 10,3+((J-1)*4);"▌
";B(J);
 290 NEXT J
 300 PRINT AT 10,27;"▌"
 302 REM ---------------------
 304 REM JEU
 310 IF E=1 THEN GOTO 1600
 320 PRINT AT 20,0;"QUEL EST VOT
RE MOUVEMENT"
 330 INPUT A
 340 IF A<1 OR A>6 THEN GOTO 320
 350 IF B(A)=0 THEN GOTO 320
 360 GOSUB 1040
 370 LET B=0
 380 FOR I=1 TO 6
 390 IF B(I)<>0 THEN LET B=1
 400 NEXT I
 410 IF B=0 THEN GOTO 3000
 420 LET B=0
 430 FOR I=8 TO 13
 440 IF B(I)<>0 THEN LET B=1
 450 NEXT I
 460 IF B=0 THEN GOTO 3000
 470 IF E=1 AND A=14 THEN GOTO 2
20
 480 IF A=7 THEN GOTO 220
 490 IF E=1 THEN GOTO 520
 500 LET E=1
 510 GOTO 220
 520 LET E=0
 530 GOTO 220
 980 REM ---------------------
 990 REM DEPLACEMENT DES PIONS
1000 LET A=J
1010 FOR Z=1 TO 14
1020 LET B(Z)=G(Z)
1030 NEXT Z
1040 LET P=B(A)
1050 LET B(A)=0
1060 FOR I=1 TO P
1070 LET A=A+1
1080 IF A=15 THEN LET A=1
1090 LET B(A)=B(A)+1
1100 NEXT I
1110 RETURN
```

```
1580 REM --------------------
1590 REM COUP DE L"ORDINATEUR
1600 PRINT AT 20,0;"JE JOUE EN
 "
1610 FOR J=1 TO 14
1620 LET G(J)=B(J)
1630 NEXT J
1640 IF B(13)=1 THEN GOTO 2500
1650 IF B(13)>=3 THEN GOTO 2500
1660 IF B(12)>=3 THEN GOTO 2520
1670 IF B(11)>=4 THEN GOTO 2540
1680 FOR J=13 TO 8 STEP -1
1690 IF G(J)=0 THEN GOTO 1770
1700 GOSUB 1000
1710 IF A=14 THEN GOTO 2600
1720 LET H(J)=A
1730 FOR U=9 TO 13
1740 LET Z=U+B(U)
1750 IF Z=14 THEN LET K(J)=K(J)+
1
1760 NEXT U
1770 NEXT J
1780 LET Z=0
1790 IF G(12)>=5 THEN GOTO 2520
1800 IF G(8)=1 THEN GOTO 2580
1810 IF G(9)=1 THEN GOTO 2560
1820 FOR J=8 TO 13
1830 LET U=K(J)
1840 IF U>Z THEN GOSUB 2620
1850 NEXT J
1860 IF Z>0 THEN GOTO 1980
1870 LET Z=8
1880 FOR J=8 TO 12
1890 LET U=H(J)
1900 IF U>Z AND U>0 THEN GOSUB 2
620
1910 NEXT J
1920 IF A>7 THEN GOTO 1980
1930 LET Z=7
1940 FOR J=8 TO 13
1950 LET U=H(J)
1960 IF U<Z AND U>0 THEN GOSUB 2
620
1970 NEXT J
1980 FOR U=8 TO 13
1990 LET K(U)=0
2000 LET H(U)=0
2010 NEXT U
2020 LET C=A-7
2030 PRINT AT 20,12;C
2040 LET J=A
2050 GOSUB 1000
2060 GOTO 370
2480 REM --------------------
2490 REM COUP CHOISI PAR ZX
2500 LET A=13
2510 GOTO 1980
2520 LET A=12
2530 GOTO 1980
2540 LET A=11
2550 GOTO 1980
2560 LET A=9
2570 GOTO 1980
2580 LET A=8
2590 GOTO 1980
2600 LET A=J
2610 GOTO 1980
2620 LET Z=U
2630 LET A=J
2640 RETURN
2980 REM -------------------
2990 REM RESULTATS
3000 IF B(14)>B(7) THEN GOTO 310
0
3010 IF B(14)<B(7) THEN GOTO 305
0
3020 PRINT AT 20,0;"NOUS SOMMES
EXAEQUO"
3030 LET E=INT (RND+.5)
3050 PRINT AT 20,0;"VOUS AVEZ GA
GNE         "
3060 LET T1=T1+1
3070 LET E=1
3080 GOTO 3200
3100 PRINT AT 20,0;"J""AI GAGNE
 "
3110 LET T2=T2+1
3120 LET E=0
3200 LET PA=PA+1
3210 PRINT AT 16,5;"ZX=";T2;TAB
(15);B$;"=";T1
3220 GOTO 170
```

LE PROGRAMME

Lignes 10 à 210

Après la saisie du prénom du joueur, on initialise le compteur de jeu, le nombre des parties gagnées ainsi que le nombre de pions par case.

Lignes 212 à 300

La variable E détermine le tour du joueur si sa valeur est zéro, et le tour du ZX81 si sa valeur est un. Au premier tour, cette variable comporte un nombre aléatoire mais, dès la deuxième partie, ce sera le perdant qui commencera.

En ligne 310, le programme se sépare en deux, suivant la valeur de E.

E=0 : TOUR DU JOUEUR

Lignes 320 à 360

Dès l'entrée du mouvement, on vérifie sa validité. En cas de mauvaise réponse, comme il n'est pas possible de passer son tour, la question est de nouveau posée jusqu'à l'obtention d'une réponse correcte.

L'appel du sous-programme de la ligne 1040 déplace les pions des cases concernées.

E=1 : TOUR DU ZX81

Lignes 1600 à 2060

Pour déterminer son choix, le ZX81 utilise trois tableaux pour comparer les solutions possibles et les plus avantageuses.

La solution trouvée est placée dans A comme pour le joueur, ce qui permet d'utiliser le sous-programme de déplacement des pions des lignes 1000 à 1110.

Lignes 1000 à 1110

On déplace les pions de la case pointée par A en les distribuant dans les cases suivantes, en considérant que la case 15 (inexistante) correspond à la case 1.

Lignes 370 à 530

Le jeu doit s'arrêter quand un des joueurs n'a plus de pion dans son terrain. Dans ces lignes, on vérifie, à chaque tour, cette éventualité.

Dans ce cas, on passe à la ligne 3000.

Lignes 3000 à 3220

Dès qu'un joueur n'a plus de point, on affiche le score de chacun, ainsi qu'un message concernant l'heureux gagnant... et le jeu continue.

DATE	NOM DU PROGRAMME : AWARI	PAGE /

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A	VARIABLE	X					Votre mouvement
B	AIGUILLAGE		X				Si B = 0 fin de partie
B$	VARIABLE	X					Votre prénom
B( )	TABLEAU		X				Nombre de pions dans chaque
							case
C	VARIABLE		X				Numéro de la case jouée par ZX
E	AIGUILLAGE		X				Si E = 1 partie ZX
							Si E = 0 partie joueur
G( )	TABLEAU		X				Sauvegarde du tableau B ( )
H( )	TABLEAU		X				Nombre de pions dans chaque
							case aprés déplacement
I	INDICE		X				Boucle FOR / NEXT
J	INDICE		X				Boucle FOR / NEXT
K( )	TABLEAU		X				Coefficient de chaque case
PA	COMPTEUR		X				Numéro de partie
T1	COMPTEUR		X				Nombre de parties gagnées par
							le Joueur
T2	COMPTEUR		X				Nombre de parties gagnées
							par ZX
U	VARIABLE		X				Coefficient de la case étudiée
Z	INDICE		X				Boucle FOR / NEXT

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Guerre des étoiles

Les vaisseaux-robots ont envahi l'espace inter-sidéral ! et il devient de plus en plus difficile d'atteindre mars. Voilà pourquoi les pilotes s'entraînent sur ce simulateur de vol.

Dès le départ (-après avoir tapé RUN-), il est indiqué, sur l'écran, le temps de vol (60 secondes maximum), le nombre de lasers utilisés (vous en avez 10 au début du jeu), et le nombre d'ennemis abattus.

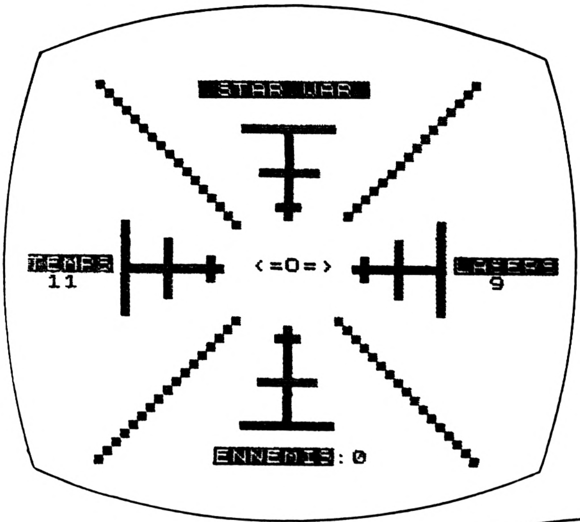
Vous appuyez sur n'importe quelle touche -sauf **BREAK**- pour tirer ; la cible change après chaque essai réussi ou non ; les compteurs de lasers et d'ennemis comptent les points en plus ou en moins ; le temps continue de s'écouler.

L'affichage des résultats s'effectue dans 3 cas :

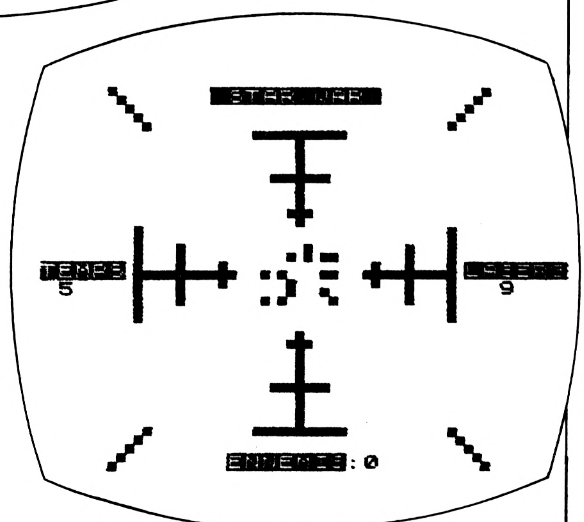
- il ne reste plus de lasers,
- les 60 secondes sont écoulées,
- vous avez éliminé 5 ennemis.

Un message apparaît avec les résultats et vous pouvez repartir.

arrêt du jeu en cours : **BREAK**







RESULTATS:

TEMPS : 60
LASERS : 4
ENNEMIS : 1

ECONOMISEZ VOS MUNITIONS ????

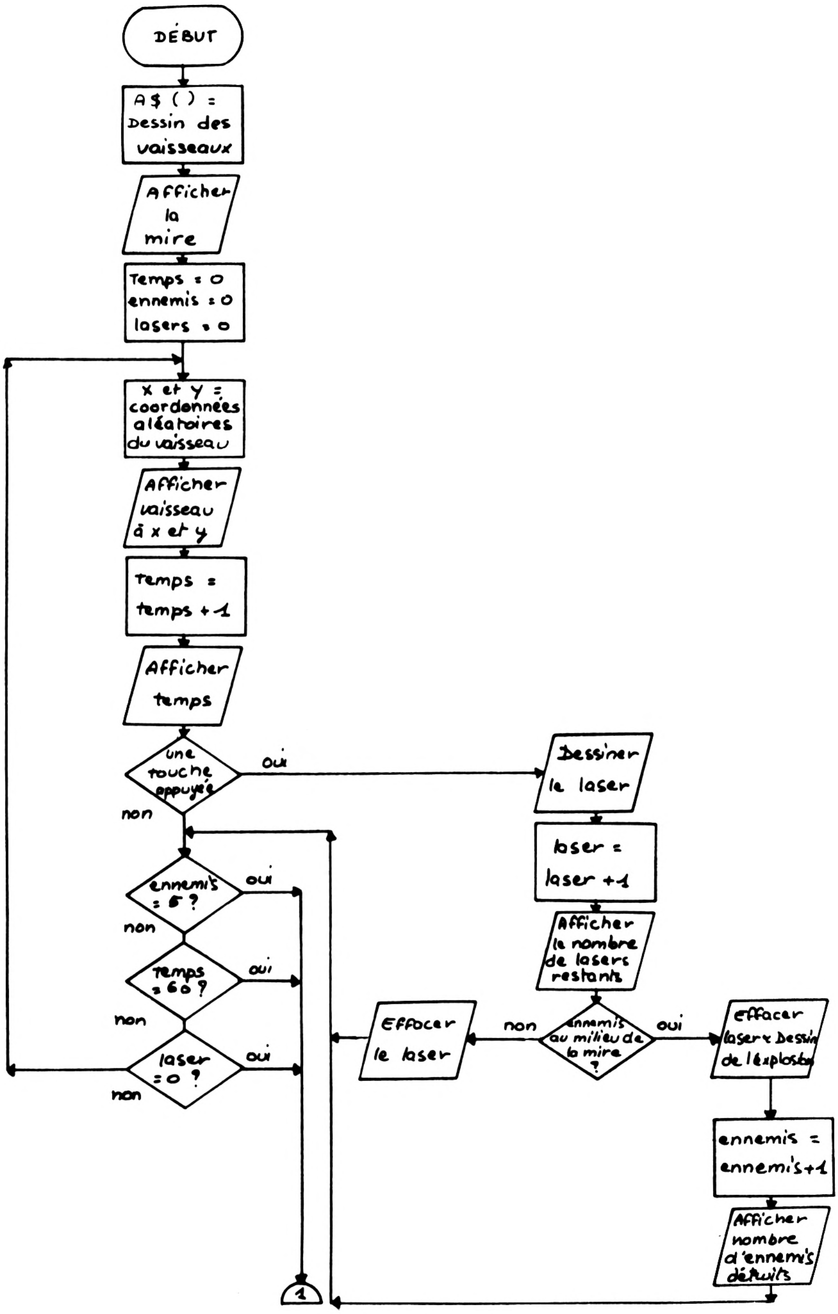
TAPEZ SUR UNE TOUCHE

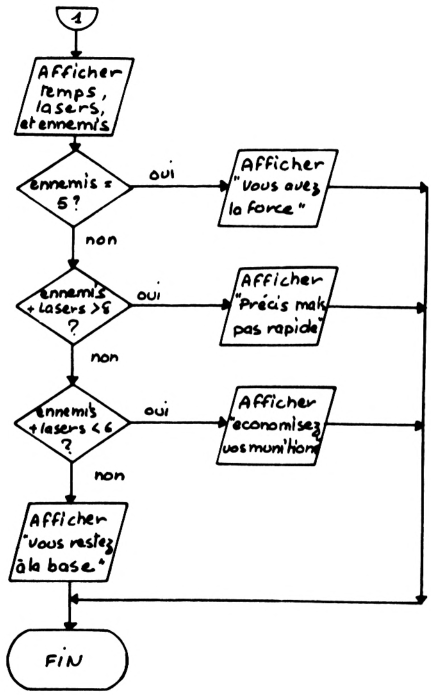
RESULTATS:

TEMPS : 60
LASERS : 9
ENNEMIS : 0

VOUS RESTEREZ A LA BASE
A LA PROCHAINE MISSION ????

TAPEZ SUR UNE TOUCHE


DÉBUT
A$ ( ) = Dessin des vaisseaux
Afficher la mire
Temps = 0
ennemis = 0
lasers = 0
x et y = coordonnées aléatoires du vaisseau
Afficher vaisseau à x et y
Temps = temps + 1
Afficher temps
une touche appuyée
oui
non
Dessiner le laser
laser = laser + 1
Afficher le nombre de lasers restants
ennemis = 5 ?
oui
non
Temps = 60 ?
oui
non
laser = 0 ?
oui
non
Effacer le laser
non
ennemis au milieu de la mire ?
oui
Effacer laser & Dessin de l'explosion
ennemis = ennemis + 1
Afficher nombre d'ennemis détruits
1


1
Afficher temps, lasers, et ennemis
ennemis = 5?
oui
Afficher "Vous avez la force"
non
ennemis + lasers > 8 ?
oui
Afficher "Précis mais pas rapide"
non
ennemis + lasers < 6 ?
oui
Afficher "economisez vos munitions"
non
Afficher "vous restez à la base"
FIN

```
   1 REM GUERRE DES ETOILES
   2 REM ------------------------
   3 CLS
   5 DIM A$(5,5)
   7 LET Z=INT (RND*5)+1
  10 LET A$(1)="<-O->"
  11 LET A$(2)="<=O=>"
  12 LET A$(3)="▀▀█▀▀"
  13 LET A$(4)="<-█->"
  14 LET A$(5)="<=█=>"
  15 PRINT AT 10,0;"TEMPS"
  16 PRINT AT 10,25;"LASERS"
  17 PRINT AT 11,27;10
  18 PRINT AT 0,10;"STAR WARS"
  19 PRINT AT 21,11;"ENNEMIS: ";0
  20 REM DESSIN DE LA MIRE
  25 FOR I=1 TO 11
  30 PLOT 30,4+I
  35 PLOT 24+I,4
  40 PLOT 30,27+I
  45 PLOT 24+I,38
  50 PLOT 11+I,22
  55 PLOT 11,16+I
  60 PLOT 37+I,22
  65 PLOT 48,16+I
  67 IF I>7 THEN GOTO 87
  70 PLOT 26+I,9
  72 PLOT 26+I,33
  74 PLOT 16,18+I
  76 PLOT 43,18+I
  77 IF I>3 THEN GOTO 87
  78 PLOT 28+I,14
  80 PLOT 28+I,29
  82 PLOT 21,20+I
  84 PLOT 39,20+I
  87 NEXT I
  88 REM ------------------------
  89 REM JEU
  90 LET T=0
  92 LET K=10
  94 LET E=0
 100 LET X=INT (RND*2)
 110 LET Y=INT (RND*5)
 200 PRINT AT 8+Y,12+X;A$(Z)
 210 LET T=T+1
 220 PRINT AT 11,1;T
 255 IF INKEY$<>"" THEN GOSUB 50
0
 270 PRINT AT 8+Y,12+X;"     "
 280 IF E=5 OR T=60 OR K=0 THEN
GOTO 900
 300 GOTO 100
 480 REM ------------------------
 490 REM DESSIN DU LASER
 500 FOR P=0 TO 22
 510 PLOT 8+P,P
 515 PLOT 8+P,43-P
 520 PLOT 52-P,P
 525 PLOT 52-P,43-P
 530 NEXT P
 540 LET K=K-1
 550 PRINT AT 11,27;K;" "
 580 IF Y=2 AND X=1 THEN GOTO 70
0
```

```
 590 PRINT AT 8+Y,12+X;A$(Z)
 600 FOR P=22 TO 0 STEP -1
 605 IF P=15 THEN PRINT AT 8+Y,1
2+X;A$(Z)
 610 UNPLOT 8+P,P
 615 UNPLOT 8+P,43-P
 620 UNPLOT 52-P,P
 625 UNPLOT 52-P,43-P
 630 NEXT P
 635 LET Z=INT (RND*5)+1
 640 RETURN
 680 REM ----------------------
 690 REM DESSIN DE L"EXPLOSION
 700 FOR P=22 TO 0 STEP -1
 710 LET W=RND*11
 720 LET V=RND*8
 730 PLOT W+24,V+18
 740 UNPLOT 8+P,P
 750 UNPLOT 8+P,43-P
 760 UNPLOT 52-P,P
 770 UNPLOT 52-P,43-P
 780 NEXT P
 790 FOR V=8 TO 12
 800 PRINT AT V,12;"        "
 810 NEXT V
 815 LET Z=INT
     (RND*5)+1
 830 LET E=E+1
 840 PRINT AT 21,19;E
 850 RETURN
 880 REM ----------------------
 890 REM RESULTATS
 900 CLS
 910 PRINT TAB 10;" RESULTATS  "
 920 PRINT
 930 PRINT " TEMPS    :  ";T
 940 PRINT " LASERS   :  ";K
 950 PRINT " ENNEMIS  :  ";E
 954 PRINT
 955 PRINT
 956 PRINT
 960 IF E=5 THEN GOTO 1000
 970 IF E+K>8 AND E>1 THEN GOTO
1020
 980 IF E+K<6 THEN GOTO 1040
 990 GOTO 1060
1000 PRINT " VOUS AVEZ LA FORCE
...."
1010 GOTO 2000
1020 PRINT " PRECIS MAIS PAS RAP
IDE ...."
1030 GOTO 2000
1040 PRINT " ECONOMISEZ VOS MUNI
TIONS ...."
1050 GOTO 2000
1060 PRINT " VOUS RESTEREZ A LA
BASE"
1070 PRINT " A LA PROCHAINE MISS
ION ...."
2000 PRINT AT 20,5;"TAPEZ SUR UN
E TOUCHE"
2010 IF INKEY$<>"" THEN RUN
2020 PRINT AT 20,5;"
              "
2030 GOTO 2000
```

LE PROGRAMME

Lignes 5 à 14

On définit, dans le tableau A$, les dessins des vaisseaux ennemis.

Lignes 15 à 87

Dessin de la mire de tir, ainsi que des compteurs de temps, lasers et ennemis.

Lignes 90 à 94

On initialise les compteurs de temps, lasers et ennemis (variables T, K et E).

Lignes 100 à 200

On attribue, aux coordonnées X et Y des vaisseaux, des valeurs aléatoires, puis on affiche le vaisseau à ces coordonnées.

Lignes 210 à 300

Après avoir incrémenté le compteur du temps (variable T), on teste si une touche du clavier a été enfoncée. Si c'est le cas, on saute au sous-programme en ligne 500. Dans le cas contraire, on efface le vaisseau ennemi en affichant des blancs aux coordonnées X et Y.

Si le nombre d'ennemis est égal à 5, ou si le temps est égal à 60, ou s'il ne reste aucun laser disponible, on considère le jeu fini et l'on va directement en ligne 900 pour les résultats.

LES SOUS-PROGRAMMES

Lignes 500 à 640

On dessine, à ce sous-programme, les quatre lasers dirigés sur le vaisseau ennemi ; on affiche, pour cela, 23 petits carrés par l'instruction **PLOT** que l'on efface après par l'instruction **UNPLOT.**

Si le vaisseau est touché, on va au sous-programme en ligne 700 pour simuler l'explosion.

Avant de retourner au programme principal, on assigne, à la variable Z, un nombre aléatoire pour changer de type de vaisseau.

Lignes 700 à 850

On simule l'explosion du vaisseau ennemi en affichant aléatoirement des carrés par l'instruction **PLOT**.

On efface ces carrés en ligne 790 en imprimant des espaces pour que le jeu continue.

On affiche le nombre d'ennemis détruits avant de revenir au jeu principal.

Lignes 900 à 2030

Une des conditions de la ligne 280 étant vraie, on affiche les résultats du jeu : temps, lasers restants et ennemis détruits. Suivant ces résultats, on affiche un message correspondant au niveau du joueur.

DATE	NOM DU PROGRAMME : GUERRE DES ETOILES	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	Programme	Cassette	Disquette	LONG.	OBSERVATIONS
A$(,)	TABLEAU		X				Dessin des vaisseaux
E	COMPTEUR		X				Nombre d'ennemis détruits
I	INDICE		X				Boucle FOR/NEXT pour le dessin
							de la mire de tir.
K	COMPTEUR		X				Nombre de lasers restants
P	INDICE		X				Boucle FOR/NEXT pour le dessin
							des lasers
T	COMPTEUR		X				Temps écoulé
V	VARIABLE		X				} Coordonnées aléatoires des
W	VARIABLE		X				} points de l'explosion d'un vaisseau
X	VARIABLE		X				} Coordonnées aléatoires
Y	VARIABLE		X				} du vaisseau
Z	VARIABLE		X				Numéro aléatoire du vaisseau

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Course automobile

N'avez-vous jamais imaginé que vous preniez l'autoroute en sens inverse ?

C'est ce qui arrive à la petite voiture de cette course folle dont le conducteur devra piloter le mieux possible pour éviter tout ce qui surgira en face de lui.

Après avoir fait **RUN**, vous choisissez les difficultés de votre épreuve en tapant pour la largeur de la route, puis pour la densité de circulation : 1, 2 ou 3.

Puis, vous tapez sur n'importe quelle touche (sauf BREAK) et le jeu se dessine sur l'écran ; vous tapez encore sur n'importe quelle touche, et ça démarre !...

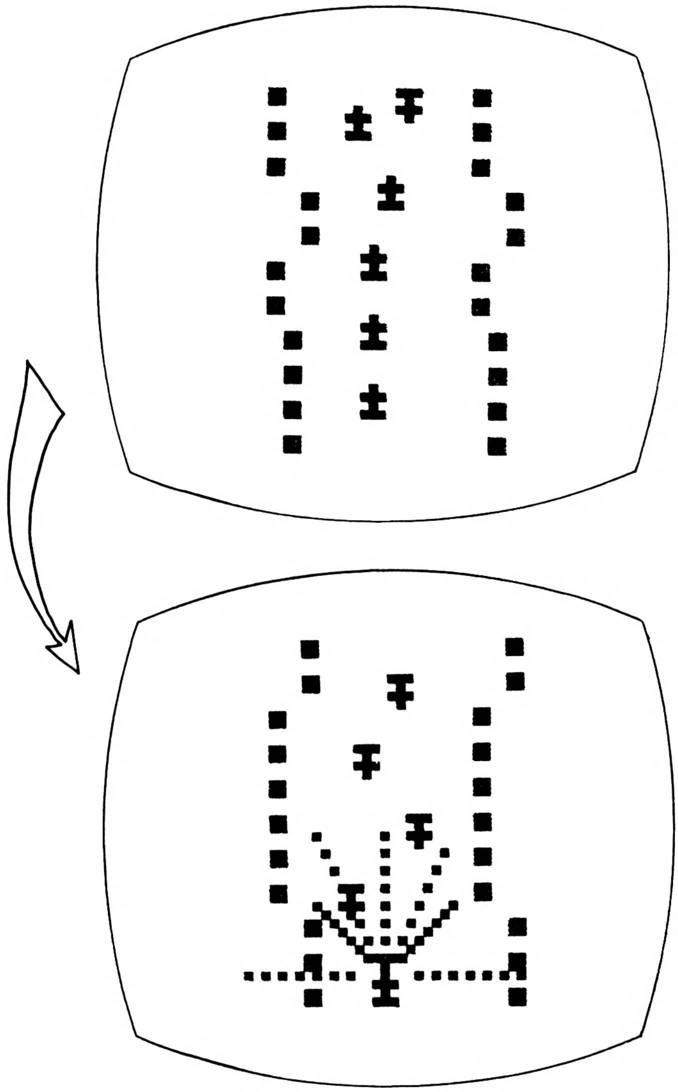
Une seule règle de conduite impérative : taper sur Q pour virer sur la gauche et sur P pour virer sur la droite.

Au bout de 5 accidents, vous découvrirez l'affichage des résultats de votre course et vous pourrez recommencer.

24H DU MANS

1....DIFFICILE
2....MOYEN
3....FACILE

LARGEUR DE LA ROUTE ?



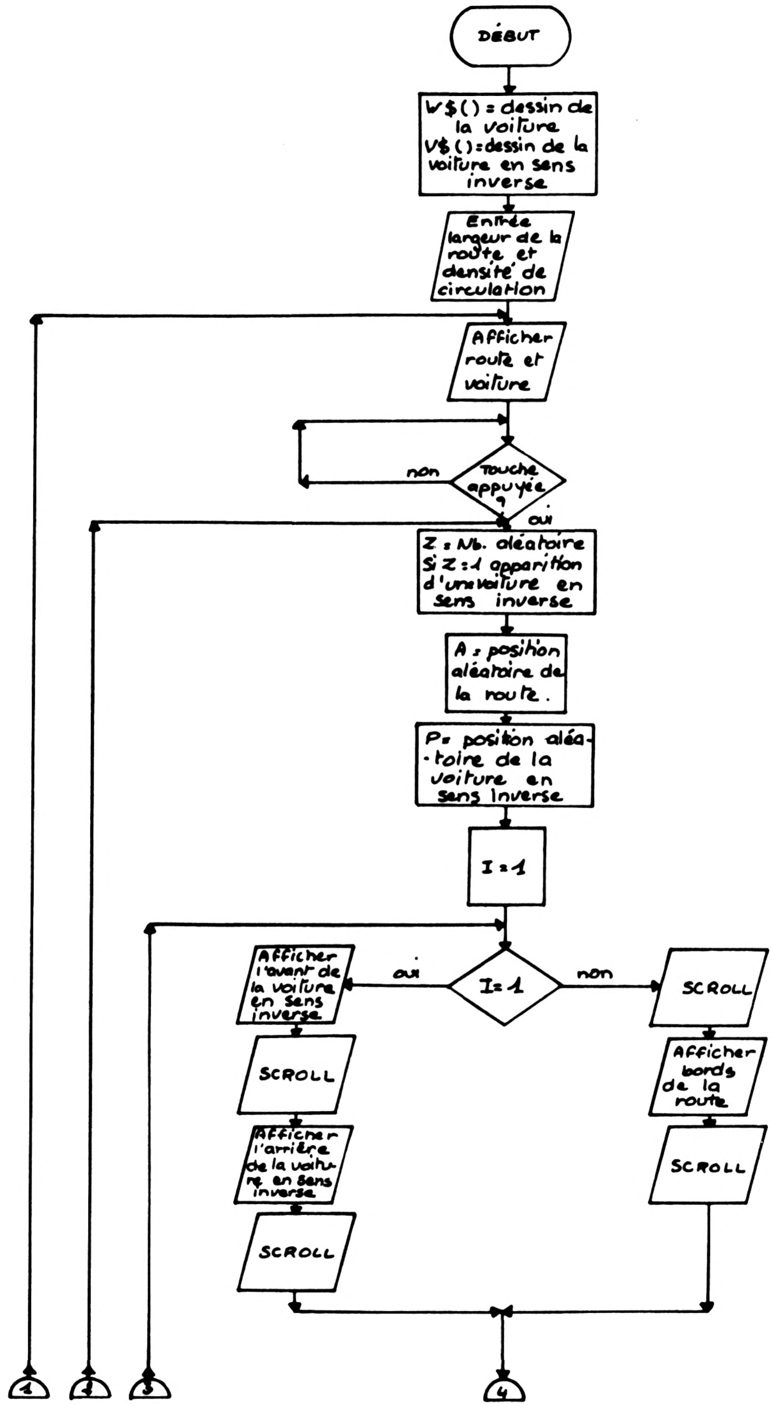
RESULTATS

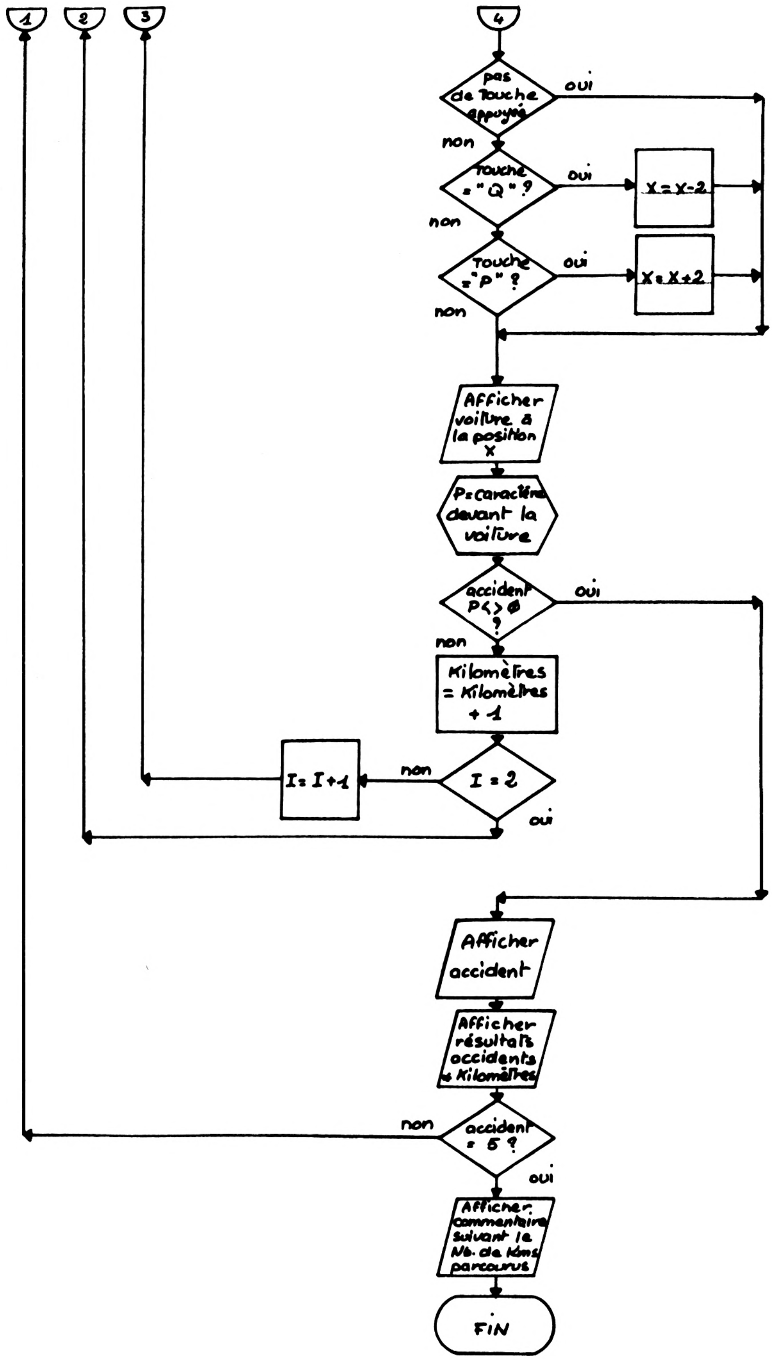
KILOMETRE :178.7

ACCIDENTS :5

* VOUS ETES UN AS *

TAPEZ SUR UNE TOUCHE


DÉBUT
V$() = dessin de la voiture
V$() = dessin de la voiture en sens inverse
Entrée largeur de la route et densité de circulation
Afficher route et voiture
Touche appuyée ?
non
oui
Z = Nb. aléatoire
Si Z = 1 apparition d'une voiture en sens inverse
A = position aléatoire de la route.
P = position aléatoire de la voiture en sens inverse
I = 1
I = 1
oui
non
Afficher l'avant de la voiture en sens inverse
SCROLL
Afficher l'arrière de la voiture en sens inverse
SCROLL
SCROLL
Afficher bords de la route
SCROLL
1
2
3
4


1
2
3
4
pas de Touche appuyée
oui
non
Touche = "Q" ?
oui
X = X-2
non
Touche = "P" ?
oui
X = X+2
non
Afficher voiture à la position X
P = caractère devant la voiture
accident P<>Ø ?
oui
non
Kilomètres = Kilomètres + 1
I = 2
non
I = I+1
oui
Afficher accident
Afficher résultats accidents & Kilomètres
accident = 5 ?
non
oui
Afficher commentaire suivant le Nb. de Kms parcourus
FIN

```
  10 REM COURSE AUTOMOBILE
 100 DIM V$(5,2,2)
 110 LET V$(1,1)=" ▄█▄"
 120 LET V$(1,2)=" ▀█▀"
 200 DIM W$(1,2,2)
 210 LET W$(1,1)=" ▀█▀"
 220 LET W$(1,2)=" ▄█▄"
 300 LET C=0
 310 LET B=0
 320 REM -----------------------
 325 REM NIVEAU DU JEU
 330 PRINT TAB 12;"[illegible]
"
 340 PRINT
 350 PRINT "1....DIFFICILE"
 360 PRINT "2....MOYEN"
 370 PRINT "3....FACILE"
 380 PRINT
 390 PRINT AT 21,0;"LARGEUR DE L
A ROUTE ?"
 400 INPUT L
 410 PRINT AT 21,0;"DENSITE DE C
IRCULATION ?"
 420 INPUT T
 470 REM -----------------------
 480 REM JEU DE DEPART
 490 CLS
 500 FOR I=1 TO 11
 510 PRINT TAB 32;" ";
 520 PRINT TAB 9;"█";TAB 20;"█";
TAB 32;" "
 530 NEXT I
 590 LET X=14
 600 PRINT AT 0,X;W$(1,1)
 610 PRINT AT 1,X;W$(1,2)
 620 LET D=(PEEK 16396+256*PEEK
16397)
 700 IF INKEY$="" THEN GOTO 700
 710 REM -----------------------
 720 REM JEU
 750 LET Z=INT (RND*T)+1
 760 LET A=INT (RND*3)+4+L
 770 LET P=INT (RND*5)+2+L
 780 LET R=11+L
 800 FOR I=1 TO 2
1000 IF I>1 THEN GOTO 1060
1010 PRINT AT 21,A+P;V$(Z,1);TAB
31;" ";
1020 SCROLL
1030 PRINT TAB A;"█";TAB A+P;V$(
Z,2);TAB A+R;"█";TAB 31;" ";
1040 SCROLL
1050 GOTO 1110
1060 PRINT AT 21,31;" ";
1070 SCROLL
```

```
1080 PRINT TAB A;"█";TAB A+R;"█"
;TAB 31;" ";
1090 SCROLL
1110 IF INKEY$="" THEN GOTO 1140
1120 IF INKEY$="Q" THEN LET X=X-
2
1130 IF INKEY$="P" THEN LET X=X+
2
1140 PRINT AT 0,X;W$(1,1)
1150 PRINT AT 1,X;W$(1,2)
1160 IF PEEK (D+67+X)<>0 OR PEEK
 (D+68+X)<>0 THEN GOTO 2000
1170 LET B=B+1
1200 NEXT I
1210 GOTO 750
1980 REM ---------------------
1990 REM ACCIDENT
2000 LET X1=X+1
2010 FOR I=2 TO 6
2020 PLOT X1*2,40-(I*2)
2030 PLOT (X1*2)+I*2,40
2040 PLOT (X1*2)-I*2,40
2050 PLOT (X1*2)+I,40-I
2060 PLOT (X1*2)-I,40-I
2070 PLOT (X1*2)+I,40-(I*2)
2080 PLOT (X1*2)-I,40-(I*2)
2090 NEXT I
2095 REM ---------------------
2096 REM RESULTATS
2100 CLS
2110 PRINT TAB 10;" RESULTATS "
2120 PRINT
2130 PRINT
2140 PRINT "KILOMETRE :";B/10
2150 PRINT
2160 PRINT
2170 LET C=C+1
2180 PRINT "ACCIDENTS :";C
2190 IF C=5 THEN GOTO 2220
2200 IF INKEY$="" THEN GOTO 2200
2210 GOTO 490
2220 PRINT
2230 PRINT
2240 IF B/C>150 THEN PRINT "   *
VOUS ETES UN AS *"
2250 IF B/C<150 AND B/C>79 THEN
PRINT "   * VOUS ETES DANGEUREUX
*"
2260 IF B/C<80 THEN PRINT "* VOU
S ETES VRAIMENT UN DANGER *
     * PUBLIC *"
2270 PRINT
2280 PRINT
2290 PRINT "TAPEZ SUR UNE TOUCHE
"
2300 IF INKEY$="" THEN GOTO 2300
2310 RUN
```

LE PROGRAMME

Lignes 100 à 310

Après déclaration des tableaux V$ et W$, on y place les dessins des voitures, puis on initialise les compteurs d'accidents et de kilomètres parcourus.

Lignes 330 à 420

On saisit au clavier les deux paramètres de difficulté, à savoir la largeur de la route et la densité de circulation ; on retrouve ces deux paramètres dans le reste du programme.

Lignes 490 à 700

Ces lignes dessinent, sur l'écran, les bords de la piste ainsi que la voiture du joueur. En ligne 700, la fonction **INKEY $** attend qu'un caractère (-autre que BREAK-) soit tapé au clavier. Il est possible d'obtenir le même résultat en écrivant **PAUSE 40 000** (ou tout autre nombre supérieur à 32 767), mais cette instruction a tendance à faire décrocher l'écran quand une touche est appuyée.

Lignes 750 à 1210

La variable Z définit l'apparition de voitures en sens inverse - W$(1) étant rempli, seul Z=1 fait apparaître une voiture. Les deux autres variables A et P positionnent respectivement la route et la voiture.

Tout le jeu est basé sur l'effet du **SCROLL** qui a pour but de faire croire au joueur que sa voiture avance parmi celles d'en face (ce qui n'est pas le cas !).

On teste, par des **INKEYS $**, les déplacements à droite ou à gauche de la voiture, tout caractère autre que P et Q étant ignoré.

Pour déterminer si un obstable se présente, on va chercher directement, dans la mémoire de l'écran, les caractères face à la voiture.

Si ceux-ci sont différents de 0 (l'espace dans le code ZX), on saute au programme qui simule l'accident.

Lignes 2000 à 2090

Elles dessinent, par l'instruction **PLOT**, l'effet du **CRASH** provoqué par la rencontre de la voiture et de l'obstacle.

Lignes 2110 à 2310

Dès la fin du dessin de l'accident, on efface l'écran pour afficher les résultats, le nombre de kilomètres parcourus, ainsi que le nombre d'accidents.

Si ceux-ci sont au nombre de 5, on affiche, à l'écran, un message correspondant à vos dons de conducteur.

DATE	NOM DU PROGRAMME : COURSE AUTOMOBILE	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A	VARIABLE		X				Position de la route
B	COMPTEUR		X				Kilomètres parcourus
C	COMPTEUR		X				Nombre d'accidents
D	VARIABLE		X				Contenu de la case mémoire
							devant la voiture.
I	INDICE		X				Boucle FOR / NEXT
L	VARIABLE	X					Choix de la largeur de la route
P	VARIABLE		X				Position de la voiture en sens
							inverse
R	VARIABLE		X				largeur de la route
T	VARIABLE	X					choix de la densité de circulation
V$( )	TABLEAU		X				Dessin voiture en sens inverse
W$( )	TABLEAU		X				Dessin de la Voiture
X	VARIABLE		X				coordonnées de la voiture
Z	AIGUILLAGE		X				Si Z = 1 apparition d'une
							voiture en sens inverse

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Chasse au trésor

Il s'agit, pour deux joueurs, d'additionner, à tour de rôle, le maximum des points inscrits sur cette grille.

Il faut, pour cela, user de beaucoup de tactique, mais il y a aussi une grande part de hasard.

Après **RUN**, un écran gris apparaît, le temps du remplissage de la grille qui s'affiche d'un seul coup.

Le joueur déplace les pions $ ou * avec les touches ← → ↑ ↓

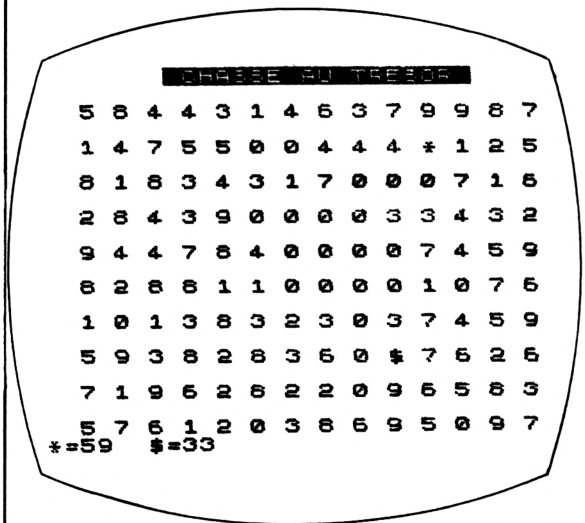
Il gagne la valeur marquée de la position où il passe ; cette position se met alors à 0. Les emplacements des 2 pions de départ ont des valeurs cachées qui se révèlent si on repasse dedans.

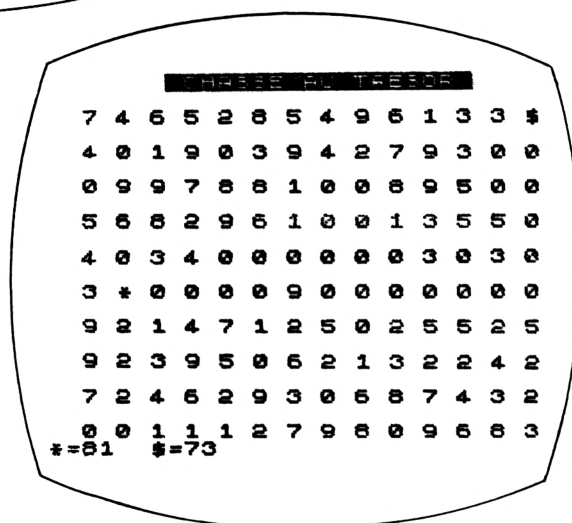
Les scores sont indiqués au bas de la grille tout au long du parcours.

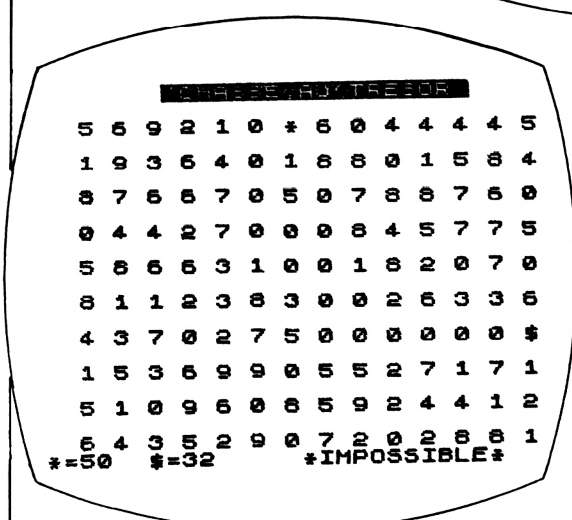
La chasse au trésor s'arrête lorsque l'un des joueurs arrive à 200 points ; il y a alors affichage des résultats.

On tape sur **NEWLINE** pour avoir une nouvelle grille.

```
   1 REM CHASSE AU TRESOR
   5 REM ----------------------
  10 REM INITIALISATION
  15 DIM A$(4,1)
  20 DIM A(22,32)
  25 DIM X(4)
  30 DIM Y(4)
  35 DIM P(4)
  36 REM ----------------------
  37 REM AFFICHAGE JEU
  40 PRINT AT 0,7;" CHASSE AU TRE
SOR "
  45 LET R=0
  49 FAST
  50 FOR Y=1 TO 10
  60 FOR X=1 TO 14
  70 LET A(Y,X)=INT (RND*10)
  80 PRINT AT Y*2,X*2;A(Y,X)
  90 NEXT X
 100 NEXT Y
 110 SLOW
 180 REM ---------------------
 190 REM JEU
 200 LET X(1)=7
 210 LET Y(1)=5
 220 LET X(2)=8
 230 LET Y(2)=6
 240 PRINT AT Y(1)*2,X(1)*2;"*"
 250 PRINT AT Y(2)*2,X(2)*2;"$"
 260 LET A$(1)="*"
 270 LET A$(2)="$"
 300 FOR I=1 TO 2
 305 LET R=0
 310 PRINT AT Y(I)*2,X(I)*2;A$(I
)
 315 IF P(I)>200 THEN GOTO 3000
 320 IF INKEY$<>"" THEN GOTO 350
 330 PRINT AT Y(I)*2,X(I)*2;" "
 340 GOTO 310
 350 GOSUB 500
 355 IF R=1 THEN GOTO 305
 360 PRINT AT 21,(I-1)*6;A$(I);"
=";P(I)
 365 NEXT I
 370 GOTO 300
 480 REM ----------------------
 490 REM DEPLACEMENT
 500 LET D$=INKEY$
 505 IF CODE D$<33 OR CODE D$>36
 THEN LET R=1
 510 IF R=1 THEN RETURN
 515 LET D=VAL D$
 520 GOSUB 1000+(D*100)
 530 IF R=1 THEN GOTO 2000
 540 PRINT AT Y(I)*2,X(I)*2;0
 550 IF D=5 THEN LET X(I)=X(I)-1
 560 IF D=6 THEN LET Y(I)=Y(I)+1
 570 IF D=7 THEN LET Y(I)=Y(I)-1
 580 IF D=8 THEN LET X(I)=X(I)+1
 590 LET P(I)=P(I)+A(Y(I),X(I))
 600 LET A(Y(I),X(I))=0
 610 PRINT AT Y(I)*2,X(I)*2;A$(I
)
 620 RETURN
1000 REM ----------------------
1010 REM VERIF DEPLACEMENT
1500 IF X(I)=1 THEN LET R=1
1510 RETURN
1600 IF Y(I)=10 THEN LET R=1
1610 RETURN
1700 IF Y(I)=1 THEN LET R=1
1710 RETURN
1800 IF X(I)=14 THEN LET R=1
1810 RETURN
1980 REM ----------------------
1990 REM MESSAGE ERREUR
2000 FOR K=1 TO 20
2010 PRINT AT 21,15;"*IMPOSSIBLE
*"
2020 PRINT AT 21,15;"
"
2030 NEXT K
2040 RETURN
2980 REM ----------------------
2990 REM FIN DE PARTIE
3000 CLS
3010 PRINT TAB 10;" RESULTATS "
3020 PRINT
3030 PRINT
3040 FOR I=1 TO 2
3050 PRINT A$(I);"   ---> ";P(I)
3060 PRINT
3070 NEXT I
3080 IF INKEY$="" THEN GOTO 3080
3090 CLS
3100 RUN
```



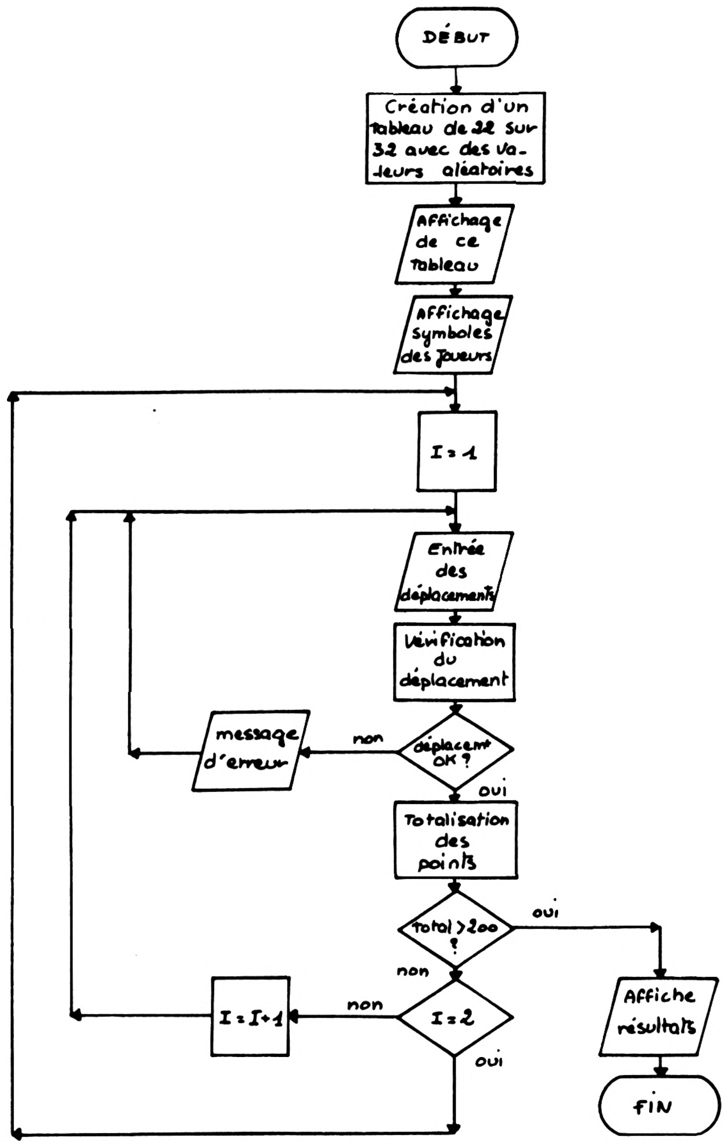




RESULTATS

* ---> 173

$ ---> 204


DÉBUT
Création d'un tableau de 22 sur 32 avec des valeurs aléatoires
Affichage de ce tableau
Affichage symboles des joueurs
I = 1
Entrée des déplacements
Vérification du déplacement
déplacem^t OK ?
non
oui
message d'erreur
Totalisation des points
total > 200 ?
oui
non
Affiche résultats
FIN
I = 2
non
oui
I = I + 1

LE PROGRAMME

Lignes 10 à 35

On initialise dans ces lignes cinq tableaux :

- A$ pour les symboles des joueurs,
- A pour les points à gagner (tableau à 2 dimensions),
- X et Y pour les coordonnées des joueurs,
- P pour le total des points par joueur.

Lignes 40 à 110

On place, dans le tableau A (10,14) des nombres aléatoires que l'on affiche sur l'écran aux coordonnées de ce même tableau. Pour plus de rapidité dans le remplissage de ce tableau, on utilise l'instruction **FAST** qui provoque l'effacement de l'écran pendant quelques secondes.

Lignes 200 à 370

On affiche successivement le symbole du joueur et un blanc pour donner un effet de clignotement.

Dès qu'une touche est appuyée, on va au sous-programme en ligne 500 pour tester si elle est valable. Au retour de ce sous-programme, on affiche les points du joueur.

SOUS-PROGRAMME

Lignes 500 à 620

On vérifie que la touche appuyée corresponde bien à une des flèches (en fait, aux touches 5 à 8).

Si ce n'est pas le cas R=1 et on retourne au programme principal.

En ligne 520, on saute à un des sous-programmes "calcule" à partir de la ligne 1500.

Au retour de ce sous-programme, si R=1, c'est qu'il n'est pas possible de se déplacer dans ce sens (sortie du cadre).

Sinon, on modifie les coordonnées du joueur et on incrémente le nombre de ses points (ligne 590).

On place, dans la case correspondante du tableau, la valeur 0.

SOUS-PROGRAMME

Lignes 2000 à 2040

Dans le cas où R=1 (déplacement impossible), en ligne 530, on affiche 20 fois le message "IMPOSSIBLE" que l'on efface aussitôt pour donner un effet de clignotement.

Lignes 3000 à 3100

Si un joueur a plus de 200 points (ligne 315), le programme efface l'écran et affiche les résultats des points obtenus par chaque joueur.

DATE	NOM DU PROGRAMME :	PAGE
	CHASSE AU TRÉSOR	1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A$( )	TABLEAU		X				Symboles des Joueurs
A( )	TABLEAU		X				Tableau des points
D$	VARIABLE	X					Déplacement du Joueur
D	VARIABLE	X					Déplacement du Joueur
I	INDICE		X				Boucle FOR / NEXT
K	INDICE		X				Boucle FOR / NEXT
P( )	TABLEAU		X				Total des points des Joueurs
R	AIGUILLAGE		X				R = 1 si erreur de déplacement
X	INDICE		X				Boucle FOR / NEXT
X( )	TABLEAU		X				} coordonnées
Y( )	TABLEAU		X				} des Joueurs
Y	INDICE		X				Boucle FOR / NEXT

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Calendrier

Comme son titre l'indique, ce programme permet d'obtenir, mois par mois, le calendrier d'une année du 20ème siècle.

En effet, après avoir tapé **RUN**, l'ordinateur demande l'année que vous désirez ; cette année doit être entrée sous la forme des deux derniers chiffres : EX=pour 1982, entrez 82, cela limite, ici, les demandes de 0 à 99, des années 1900 à 1999.

Après avoir enregistré l'année, l'ordinateur en présente le mois de janvier sur l'écran. Il suffit d'appuyer sur n'importe quelle touche pour voir apparaître ensuite le mois de février, etc...

Ce qui permet, entre chaque nouveau mois, de faire une **COPY** sur l'imprimante et d'obtenir, ainsi, un calendrier complet de l'année.

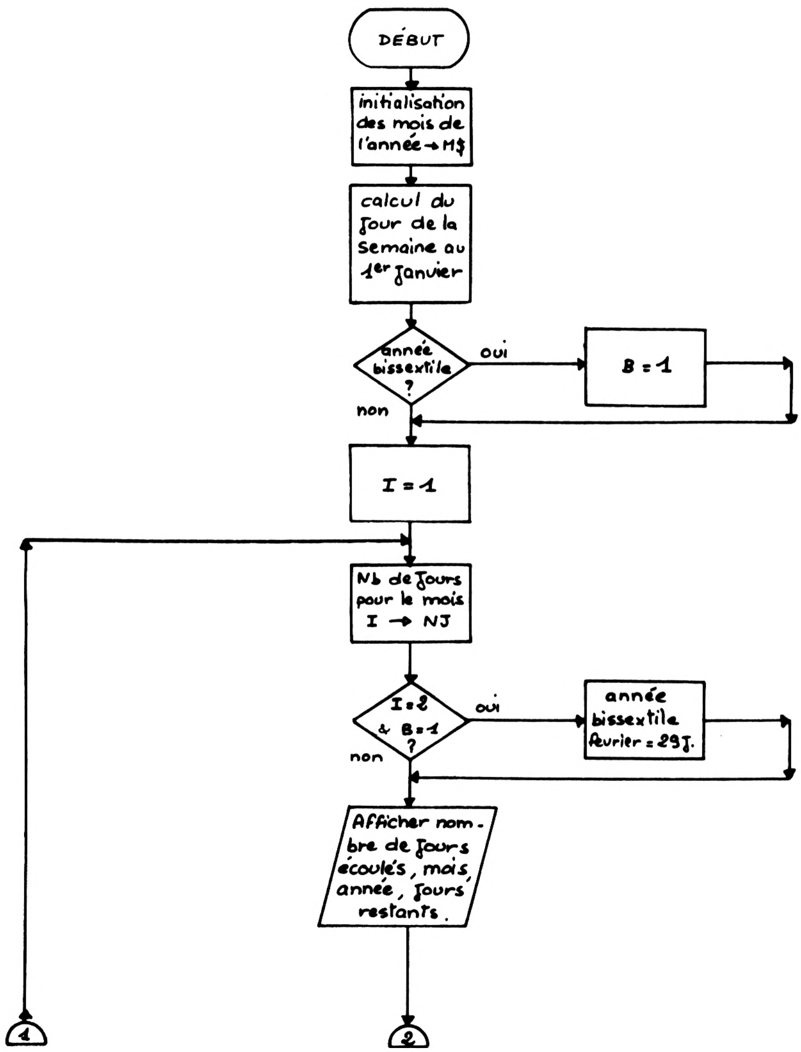
181	JUILLET			82	184	
L	M	M	J	V	S	D
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

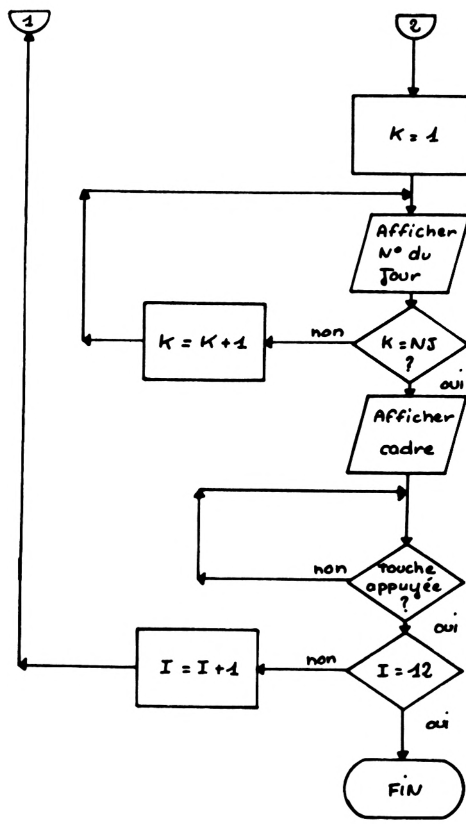
212	AOUT			82	153	
L	M	M	J	V	S	D
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

304	NOVEMBRE			82	61	
L	M	M	J	V	S	D
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

334	DECEMBRE			82	31	
L	M	M	J	V	S	D
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		




DÉBUT
initialisation des mois de l'année → M$
calcul du jour de la semaine au 1er janvier
année bissextile ?
oui
B = 1
non
I = 1
Nb de jours pour le mois I → NJ
I = 2 & B = 1 ?
oui
année bissextile février = 29 J.
non
Afficher nombre de jours écoulés, mois, année, jours restants.
1
2


1
2
K = 1
Afficher
N° du
Jour
K = NJ
?
non
K = K+1
oui
Afficher
cadre
touche
appuyée
?
non
oui
I = 12
non
I = I+1
oui
FIN

```
  10 REM CALENDRIER
  20 REM --------------------
  30 DIM M$(132)
  40 LET B=0
  50 LET J=0
 100 LET M$="JANVIER   31FEVRIER
 28MARS      31AVRIL    30MAI
   31JUIN       30JUILLET  31AOUT
    31SEPTEMBRE30OCTOBRE   31NOVEM
BRE 30DECEMBRE 31"
 180 REM --------------------
 190 REM JOUR DE LA SEMAINE
 200 PRINT AT 21,0;"ANNEE DU CAL
ENDRIER ?"
 210 INPUT U
 220 LET P=(U-1)/100
 230 LET Q=INT (P/4)
 240 LET M=36+INT ((5*U)/4)-INT
(P)+Q
 250 LET Y=M-(7*INT (M/7))
 260 IF INT (U/4)=U/4 THEN LET B
=1
 280 REM --------------------
 290 REM CALENDRIER
 300 FOR I=1 TO 12
 310 LET NJ=VAL (M$(I*11-1 TO I*
11))
 320 IF I=2 AND B=1 THEN LET NJ=
29
 330 CLS
 350 PRINT "▄▄▄▄ ";J;" ▄▄▄";M$(I*1
1-10 TO I*11-2);" ";U;"▄▄▄ ";365+
B-J;TAB 29;"▛▀▀▀"
 360 PRINT "▌";TAB 31;"█"
 370 PRINT "▌   L   M   M   J
U   S   D   █"
 380 PRINT
 390 PRINT
 400 FOR K=1 TO NJ
 410 PRINT TAB Y*4;K;
 420 LET Y=Y+1
 430 IF Y=8 THEN PRINT
 440 IF Y=8 THEN PRINT
 450 IF Y=8 THEN LET Y=1
 460 NEXT K
 470 LET J=J+NJ
 500 FOR L=0 TO 31
 510 PRINT AT 3,L;"▄"
 520 PRINT AT 19,L;"▄"
 530 NEXT L
 540 FOR L=3 TO 19
 550 PRINT AT L,0;"▌"
 560 PRINT AT L,31;"▐"
 570 IF L=3 OR L=19 THEN PRINT A
T L,31;"▟"
 580 IF L=3 OR L=19 THEN PRINT A
T L,0;"▙"
 590 NEXT L
 600 IF INKEY$="" THEN GOTO 600
 610 NEXT I
 620 CLS
 630 RUN
```

LE PROGRAMME

Lignes 30 à 100

Le tableau A$ contient la liste des noms des mois de l'année ainsi que le nombre de jours par mois.

Cette formule permet de palier au manque des instructions **DATA** et **READ**.

Lignes 200 à 260

On saisit l'année du calendrier dans la variable U. Cette variable est alors utilisée pour connaître le numéro du jour de la semaine du ler janvier de cette année (lundi=1, mardi=2, etc...).

En ligne 260, on regarde s'il s'agit d'une année bissextile ; si c'est le cas, on met la variable B à 1.

Lignes 300 à 630

Pour chaque mois de l'année, on extrait, du tableau A$, le nombre de jours par mois (variable NJ).

S'il s'agit du mois de février, on vérifie le contenu de la variable B (année bissextile).

La ligne 350 affiche, à l'écran, le nombre de jours écoulés (variable J), le nom du mois (on cherche celui-ci dans le tableau A$), l'année étudiée et le nombre de jours restants.

La variable Y donnant la position du jour suivant, on inscrit le numéro de ce jour en utilisant l'instruction **TAB (Y)**.

Quand Y=8 (le 8ème jour de la semaine), on remet cette variable à 1, et ainsi de suite.

En ligne 470, on incrémente le nombre de jours écoulés du nombre de jours que l'on vient d'afficher.

Les lignes 500 à 590 dessinent un cadre autour du calendrier.

DATE	NOM DU PROGRAMME : CALENDRIER	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
B	AIGUILLAGE		X				B=1 si année bissextile
I	INDICE		X				Boucle FOR / NEXT pour
							l'édition des 12 mois de l'année
J	COMPTEUR		X				Nombre de Jours écoulés
K	INDICE		X				Boucle FOR / NEXT pour
							l'édition des Jours
L	INDICE		X				Boucle FOR / NEXT pour le
							dessin du cadre
M$( )	TABLEAU		X				Liste des mois et du nombre
							de Jours par mois.
M	VARIABLE		X				Utiliseé pour le calcul du Jour
							de la semaine.
NJ	VARIABLE		X				Nombre de Jours par mois
P	VARIABLE		X				} Utilisées pour le calcul du
Q	VARIABLE		X				} Jour de la semaine
U	VARIABLE	X					Anneé à étudier
Y	VARIABLE		X				Numero du Jour de la semaine :
							Lundi = 1 , mardi = 2 , ...

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Tiercé

Et maintenant, les paris sont ouverts !

Huit joueurs peuvent suivre cette course à Auteuil ; le programme demande d'ailleurs, dès le début, le nombre de joueurs, de manière à prévoir la tenue des paris.

Puis, vous notez les noms des parieurs (pas plus de 7 lettres par nom). Après cela, vous donnez, pour chaque joueur, une combinaison des numéros de trois chevaux sur 7 partants.

Les résultats de ce tiercé seront :

0 point pour aucun cheval arrivé dans les 3 premiers,
0 point pour 1 cheval arrivé dans les 3 premiers,
2 points pour 2 chevaux arrivés dans les 3 premiers,
5 points pour 3 chevaux arrivés dans le désordre,
10 points pour le tiercé dans l'ordre.

A la fin de vos paris, tapez sur n'importe quelle touche, et le terrain de course apparaît. Vous déclenchez le jeu en tapant sur **NEWLINE**.

La barrière descend et la course NO 1 est partie...

A la fin de la course, l'affichage de l'arrivée se fait au bas de l'écran.

Pour avoir les résultats complets par joueur, on tape sur **NEWLINE** ; de même, pour refaire une autre course.

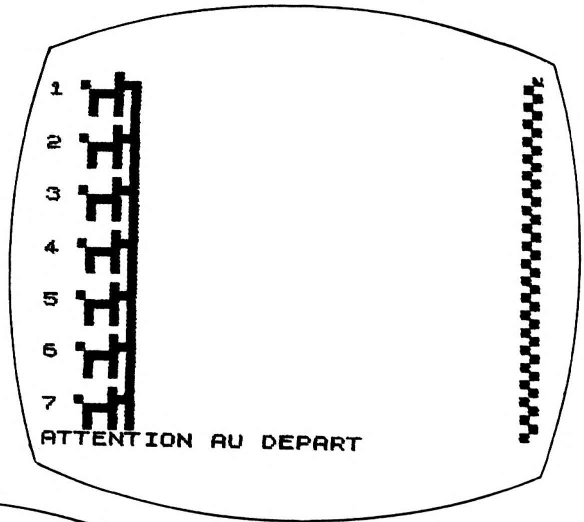
COURSE : 1

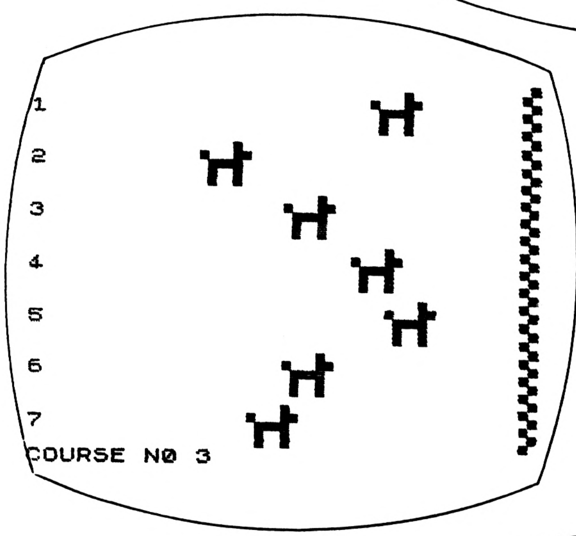
CH.1 = 1

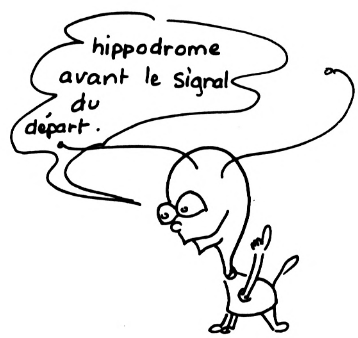
CH.2 = 3

CH.3 = 4

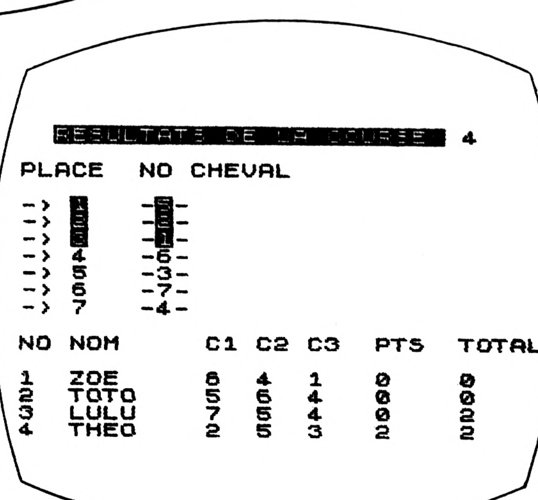
COMBINAISON DE JEAN-F
CHEVAL 3

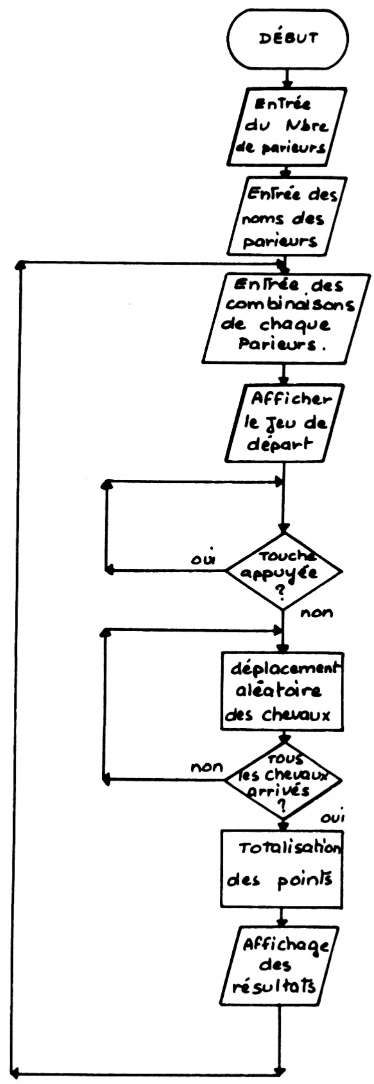











DÉBUT
Entrée du Nbre de parieurs
Entrée des noms des parieurs
Entrée des combinaisons de chaque Parieurs.
Afficher le Jeu de départ
Touche appuyée ?
oui
non
déplacement aléatoire des chevaux
Tous les chevaux arrivés ?
non
oui
Totalisation des points
Affichage des résultats

```
1 REM TIERCE
3 LET C=0
4 GOSUB 4000
5 GOSUB 6000
7 LET R=0
10 DIM X(7)
15 DIM R(7)
20 FOR I=1 TO 7
25 LET R(I)=0
30 LET X(I)=1
40 LET Y=(I-1)*3
50 GOSUB 2000
60 PRINT AT Y+1,0;I
70 NEXT I
80 FOR I=0 TO 20
90 PRINT AT I,31;"▞"
100 NEXT I
110 FOR I=2 TO 41
115 PLOT 10,I
120 NEXT I
125 REM ----------------------
126 REM JEU
130 PRINT AT 21,0;"ATTENTION AU
DEPART"
140 IF INKEY$="" THEN GOTO 140
150 PRINT AT 21,0;"COURSE N0 ";
C;" "
160 FOR I=1 TO 20
170 PRINT AT I,5;" "
180 NEXT I
200 LET I=INT (RND*7)+1
210 LET Y=(I-1)*3
220 LET X(I)=X(I)+1
300 GOSUB 1000
310 IF R<7 THEN GOTO 200
320 PRINT AT 21,0;"RESULTAT "
330 IF INKEY$="" THEN GOTO 330
340 GOSUB 7000
350 CLS
360 GOTO 5
980 REM ----------------------
990 REM DEPLACEMENT DES CHEVAUX
1000 IF X(I)>27 THEN RETURN
2000 PRINT AT Y+0,X(I);"   ▄ "
2010 PRINT AT Y+1,X(I);" ▀█▀▀"
2020 PRINT AT Y+2,X(I);"  █ █"
2030 IF X(I)=27 THEN GOTO 3000
2040 RETURN
2980 REM ---------------------
2990 REM VERIFICATION ARRIVEE
3000 IF R=0 THEN PRINT AT 21,0;"
ARRIVEE: -"
3020 LET R=R+1
3030 PRINT AT 21,8+(R*2);I;"-"
3040 LET R(R)=I
3050 RETURN
3930 REM ---------------------
3990 REM SAISIE JOUEURS
4000 PRINT TAB 10;"COURSE A [illegible]"
4010 PRINT
4020 PRINT AT 18,0;"NB DE PARIEU
RS ?"
4030 GOSUB 5000
4035 INPUT PA
4040 IF PA>8 THEN GOTO 4020
4045 DIM C(PA,3)
4050 DIM N$(PA,7)
4055 DIM P(PA)
4060 FOR I=1 TO PA
4070 PRINT AT 18,0;"NOM DU PARIE
UR ";I;" ?"
4080 INPUT N$(I)
4090 NEXT I
4100 RETURN
```

```
4980 REM ----------------------
4990 REM FLECHE DES INPUT
5000 PRINT AT 19,0;" █ "
5010 PRINT AT 20,0;"▄█▄"
5020 PRINT AT 21,0;" ▀ "
5030 RETURN
5980 REM ----------------------
5990 REM SAISIE DES PARIS
6000 CLS
6005 LET C=C+1
6010 PRINT TAB 10;"COURSE :";C
6020 GOSUB 5000
6030 FOR I=1 TO PA
6032 PRINT AT 10,0;"
               "
6035 FOR J=1 TO 3
6040 PRINT AT 18,0;"COMBINAISON
DE ";N$(I);" CHEVAL ";J
6050 INPUT C(I,J)
6052 PRINT AT 10,(J-1)*10;"CH.";
J;" = ";C(I,J)
6055 NEXT J
6060 NEXT I
6070 CLS
6080 RETURN
6980 REM ----------------------
6990 REM RESULTATS
7000 CLS
7010 PRINT "  RESULTATS DE LA CO
URSE  ";C
7020 PRINT
7030 PRINT "PLACE  NO CHEVAL"
7035 PRINT
7040 FOR I=1 TO 7
7042 IF I>3 THEN GOTO 7050
7046 PRINT "-> ";CHR$ (I+156);"
   -";CHR$ (R(I)+156);"-"
7047 GOTO 7060
7050 PRINT "-> ";I;"    -";R(I);"
-"
7060 NEXT I
7062 PRINT
7064 PRINT "NO NOM       C1 C2 C3
 PTS  TOTAL"
7066 PRINT
7070 FOR I=1 TO PA
7072 LET P=0
7074 LET T=0
7076 LET TO=0
7080 FOR J=1 TO 3
7090 FOR K=1 TO 3
7100 IF R(J)=C(I,K) THEN LET P=P
+1
7105 IF J=K AND R(J)=C(I,K) THEN
 LET T=T+1
7110 NEXT K
7120 NEXT J
7130 IF T=3 THEN LET TO=TO+5
7140 IF P=3 THEN LET TO=TO+5
7150 IF P=2 THEN LET TO=TO+2
7160 LET P(I)=P(I)+TO
7180 PRINT I;" ";N$(I);" ";C(I,
1);" ";C(I,2);" ";C(I,3);" "
;TO;" ";P(I)
7190 NEXT I
7200 IF INKEY$="" THEN GOTO 7200
7210 RETURN
```

Lignes 3 à 120

Cette partie du programme construit le jeu du départ. Après initialisation des principales variables (voir identification des variables), le sous-programme en 4000 saisit le nombre de joueurs et leurs noms. Pour que les entrées soient plus faciles, le sous-programme de la ligne 5000 dessine une flèche désignant l'endroit de l'écran où doivent être tapées les données. Puis, c'est la saisie des paris du sous-programme des lignes 6000 à 6080. Comme précédemment, on dessine une flèche pointant les entrées par le clavier.

Le jeu de départ utilise le sous-programme des lignes 2000 pour dessiner les chevaux.

Lignes 130 à 360

Après l'enfoncement d'une touche, c'est le jeu qui commence. On tire aléatoirement en ligne 200 le numéro du cheval qui doit avancer d'une case. Le cheval choisi est déplacé par le sous-programme des lignes 1000 à 2040.

Dans le cas où un cheval est sur la ligne d'arrivée, le sous-programme en 3000 affiche son numéro et incrémente le nombre de chevaux arrivés.

Dès que les sept chevaux sont sur la ligne d'arrivée, le sous-programme de la ligne 7000 affiche le tableau d'arrivée, ainsi que les points gagnés par chaque parieur.

Après consultation de ce tableau, on retourne à la ligne 5 pour une nouvelle course.

DATE	NOM DU PROGRAMME : TIERCÉ	PAGE /

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	Clavier	Programme	Cassette	Disquette	LONG.	OBSERVATIONS
C	COMPTEUR		X				Numéro de la course
C ( )	TABLEAU	X					Tableau contenant les paris
I	COMPTEUR		X				Numéro du cheval à avancer
J	INDICE		X				Boucle FOR / NEXT
N$ ( )	TABLEAU	X					Nom des parieurs
P	COMPTEUR		X				Nombre de chevaux dans l'ordre
P ( )	TABLEAU		X				Tableau contenant les points des joueurs
PA	VARIABLE	X					Nombre de parieurs
R	COMPTEUR		X				Numéro d'arrivée
R ( )	TABLEAU		X				Tableau d'arrivée
T	COMPTEUR		X				Nombre de chevaux dans le désordre
TO	COMPTEUR		X				Total des points
X ( )	TABLEAU		X				} positions des
Y ( )	TABLEAU		X				} chevaux

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Fusée lunaire

Un navigateur spatial s'entraîne en conduisant une fusée d'un côté à l'autre d'une montagne lunaire.

Au début du jeu, la montagne se dessine sur l'écran, ainsi que la fusée et l'endroit opposé où elle doit retomber ; sont aussi indiquées la quantité d'oxygène initialisée à 20 et la quantité de carburant initialisée à 90.

Le joueur doit donner la hauteur et la largeur du déplacement en deux temps chaque fois ; il doit :

- préciser le sens positif ou négatif ;
- donner l'amplitude de 1 à 9.

L'oxygène et le carburant diminuent à chaque coup.

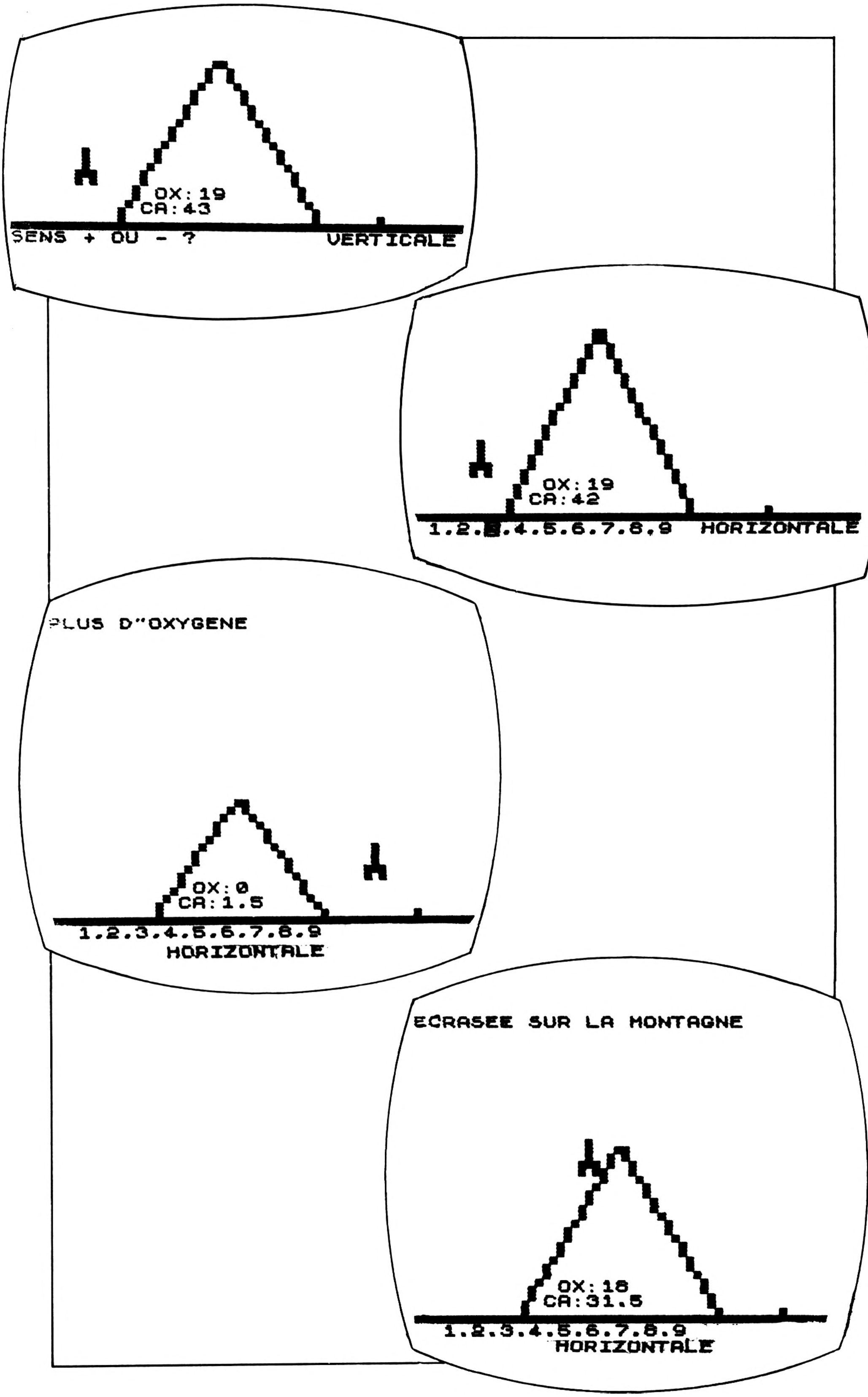
En cas d'erreur de navigation, 6 messages peuvent apparaître :

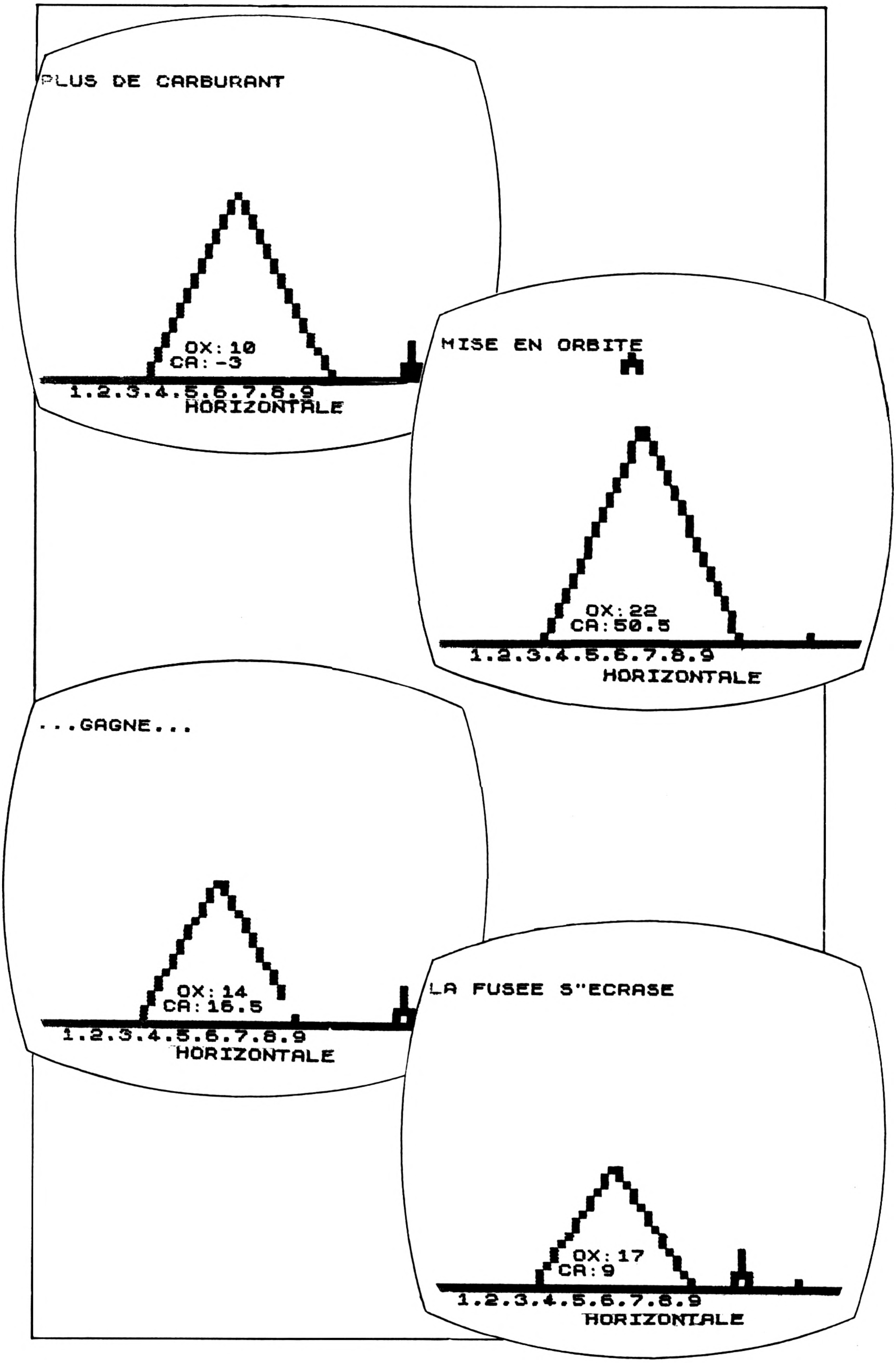
- ECRASEE SUR LA MONTAGNE,
- SORTIE DE L'ESPACE,
- MISE EN ORBITE,
- LA FUSEE S'ECRASE,
- PLUS DE CARBURANT,
- PLUS D'OXYGENE.

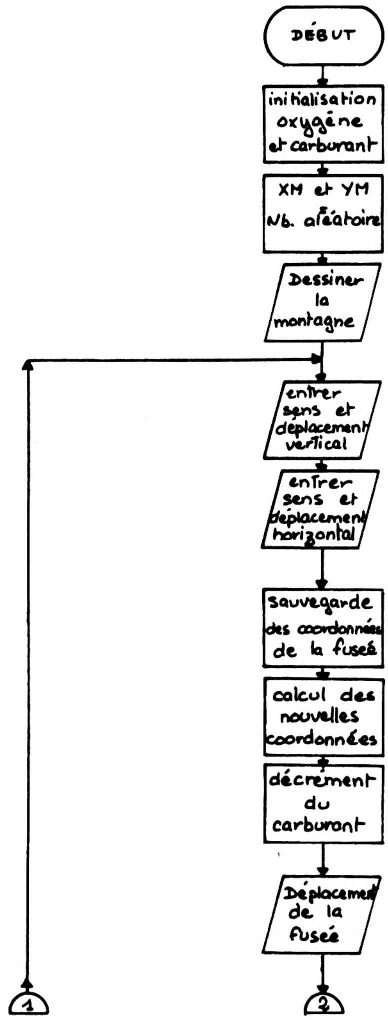
Des paramètres précis régissent les rapports des données entrées par le joueur. Au cas où il ne parviendrait jamais à poser sa fusée, il pourrait jeter un coup d'oeil au programme...

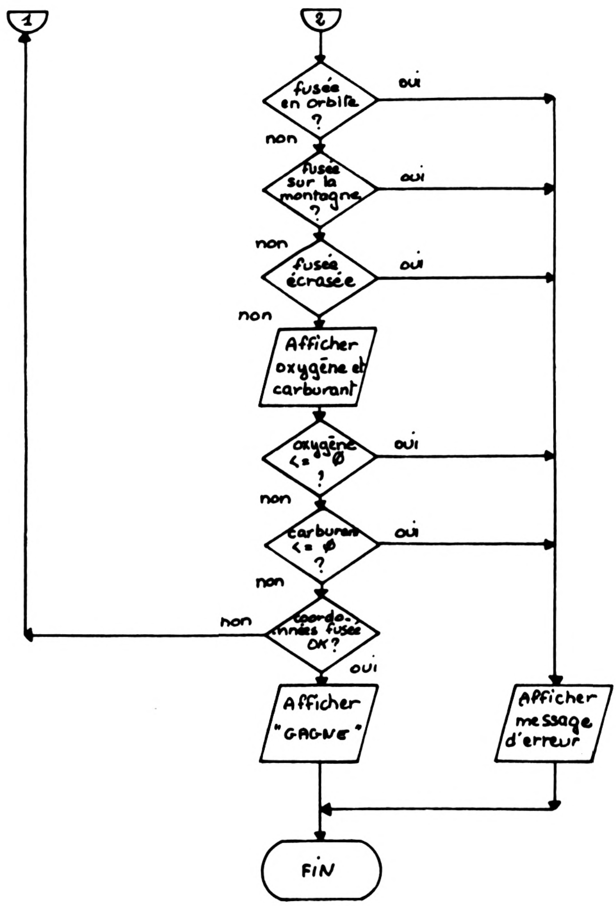
Sinon, le message "GAGNE" apparaîtra ; l'oxygène et le carburant se réinitialiseront.

Dans tous les cas, un nouveau jeu se redessine tout seul, après quelques instants.


OX:19
CA:43
SENS + OU - ?
VERTICALE
OX:19
CA:42
1.2.3.4.5.6.7.8.9
HORIZONTALE
PLUS D"OXYGENE
OX:0
CA:1.5
1.2.3.4.5.6.7.8.9
HORIZONTALE
ECRASEE SUR LA MONTAGNE
OX:18
CA:31.5
1.2.3.4.5.6.7.8.9
HORIZONTALE


PLUS DE CARBURANT
OX:10
CA:-3
1.2.3.4.5.6.7.8.9
HORIZONTALE
MISE EN ORBITE
OX:22
CA:50.5
1.2.3.4.5.6.7.8.9
HORIZONTALE
...GAGNE...
OX:14
CA:16.5
1.2.3.4.5.6.7.8.9
HORIZONTALE
LA FUSEE S"ECRASE
OX:17
CA:9
1.2.3.4.5.6.7.8.9
HORIZONTALE


DÉBUT
initialisation oxygène et carburant
XM et YM Nb. aléatoire
Dessiner la montagne
entrer sens et déplacement vertical
entrer sens et déplacement horizontal
sauvegarde des coordonnées de la fusée
calcul des nouvelles coordonnées
décrément du carburant
Déplacement de la fusée
1
2


1
2
fusée en orbite ?
oui
non
fusée sur la montagne ?
oui
non
fusée écrasée
oui
non
Afficher oxygène et carburant
oxygène <= Ø ?
oui
non
carburant <= Ø ?
oui
non
coordonnées fusée OK ?
non
oui
Afficher "GAGNE"
Afficher message d'erreur
FIN

```
  10 REM FUSEE LUNAIRE
  20 LET APV=0
  30 LET APH=0
  40 LET OX=20
  50 LET CA=90
  80 REM ----------------------
  90 REM DESSIN DE LA MONTAGNE
 100 LET XM=20+(RND*16)
 110 LET YM=16+(RND*18)
 200 FOR I=0 TO YM
 210 PLOT 14+(((XM/2)/YM)*I),I+2
 220 PLOT XM+14-(((XM/2)/YM)*I),
I+2
 230 NEXT I
 240 FOR I=0 TO 63
 250 PLOT I,2
 260 NEXT I
 270 PLOT 51,3
 300 LET XF=7
 310 LET AXF=7
 320 LET YF=5
 330 LET AYF=3
 340 GOSUB 2000
 345 REM ---------------------
 346 REM JEU
 350 PRINT AT 21,21;" VERTICALE
"
 360 GOSUB 2500
 370 LET PV=0
 400 PRINT AT 21,21;"HORIZONTALE
"
 410 GOSUB 2500
 420 LET PH=0
 430 LET DXF=XF
 440 LET DYF=YF
 500 LET XF=XF+PH+APH
 510 LET YF=YF+PV+APV
 520 LET APV=PV/2
 530 LET APH=PH/2
 540 IF PV>0 THEN GOTO 570
 550 LET CA=CA-PH+PV
 560 GOTO 700
 570 LET CA=CA-PH-PV
 700 GOSUB 4000
 750 LET OX=OX-1
 760 PRINT AT 18,10;"OX:";OX;"  "
 770 PRINT AT 19,9;"CA:";CA;"   "
 780 IF OX<=0 THEN GOTO 5080
 790 IF CA<=0 THEN GOTO 5100
 800 IF INT (XF)=50 AND INT (YF)
=3 THEN GOTO 6000
 900 GOTO 350
1980 REM ---------------------
1990 REM AFFICHE LA FUSEE
2000 IF YF>=40 THEN GOTO 5000
2001 IF YF<3 THEN GOTO 5020
2002 IF XF>=56 OR XF<3 THEN GOTO
 5040
2005 UNPLOT AXF,AYF
2010 UNPLOT AXF+2,AYF
2020 UNPLOT AXF,AYF+1
2030 UNPLOT AXF+2,AYF+1
2040 UNPLOT AXF+1,AYF+1
2050 UNPLOT AXF+1,AYF+2
2060 UNPLOT AXF+1,AYF+3
2070 UNPLOT AXF+1,AYF+4
2100 PLOT XF,YF
2110 PLOT XF+2,YF
2120 PLOT XF,YF+1
2130 PLOT XF+2,YF+1
2140 PLOT XF+1,YF+1
2150 PLOT XF+1,YF+2
2160 PLOT XF+1,YF+3
2170 PLOT XF+1,YF+4
2180 LET AXF=XF
2190 LET AYF=YF
2200 IF XF+1>(XM/2)+14 THEN GOTO
 2230
2210 IF XF+3>=14+(((XM/2)/YM)*(Y
F-2)) THEN GOTO 5060
2220 GOTO 2300
2230 IF XF-1<=XM+14-(((XM/2)/YM)
*(YF-2)) THEN GOTO 5060
2300 RETURN
```

```
2480 REM ---------------------
2490 REM CHOIX Q. CARBURANT
2500 PRINT AT 21,0;"SENS + OU -
?       "
2505 IF INKEY$="" THEN GOTO 2505
2510 IF INKEY$<>"+" AND INKEY$<>
"-" THEN GOTO 2500
2520 LET S$=INKEY$
3000 PRINT AT 21,0;"  1.2.3.4.5.
6.7.8.9";
3030 LET Q=5
3035 IF INT (Q)<>Q THEN GOTO 305
0
3040 PRINT AT 21,Q*2;CHR$ (Q+156
)
3045 GOTO 3055
3050 PRINT AT 21,Q*2;"■"
3055 IF INKEY$="" THEN GOTO 3055
3060 IF INT (Q)<>Q THEN GOTO 307
5
3065 PRINT AT 21,Q*2;CHR$ (Q+28)
3070 GOTO 3080
3075 PRINT AT 21,Q*2;"."
3080 IF INKEY$=CHR$ 118 THEN GOT
O 3200
3085 IF Q=1 THEN GOTO 3095
3090 IF INKEY$="Q" THEN LET Q=Q-
0.500
3095 IF Q=9 THEN GOTO 3110
3100 IF INKEY$="P" THEN LET Q=Q+
.5
3104 IF Q<0.5 THEN LET Q=0
3105 REM PRINT AT 21,28;Q
3110 GOTO 3035
3200 IF S$="-" THEN LET Q=Q*-1
3210 RETURN
3980 REM --------------------
3990 REM DEPLACEMENT FUSEE
4000 IF YF-DYF=0 THEN GOTO 4003
4001 LET INC=(XF-DXF)/(YF-DYF)
4002 GOTO 4005
4003 LET INC=XF-DXF
4005 IF DXF<XF AND INC<0 THEN LE
T INC=INC*(-1)
4010 IF DXF>XF AND INC>0 THEN LE
T INC=INC*(-1)
4015 LET XF=DXF-INC
4020 LET ST=1
4025 IF DYF>YF THEN LET ST=-1
4030 FOR I=DYF TO YF STEP ST
4035 LET YF=I
4040 LET XF=XF+INC
4050 GOSUB 2000
4060 NEXT I
4100 RETURN
4980 REM ---------------------
4990 REM PERDU
5000 LET M$="MISE EN ORBITE"
5010 GOTO 5200
5020 LET M$="LA FUSEE S""ECRASE"
5030 GOTO 5200
5040 LET M$="SORTIE DE L""ESPACE
"
5050 GOTO 5200
5060 LET M$="ECRASEE SUR LA MONT
AGNE"
5070 GOTO 5200
5080 LET M$="PLUS D""OXYGENE"
5090 GOTO 5200
5100 LET M$="PLUS DE CARBURANT"
5200 FOR I=1 TO 30
5210 PRINT AT 0,0;M$
5220 PRINT AT 0,0;"
          "
5230 NEXT I
5240 CLS
5250 RUN
5980 REM --------------------
5990 REM GAGNE
6000 PRINT AT 0,0;"...GAGNE..."
6010 FOR I=0 TO 20
6020 PRINT AT 18,13;I
6030 NEXT I
6040 FOR I=0 TO 90 STEP 5
6050 PRINT AT 19,12;I
6060 NEXT I
6070 CLS
6080 RUN
```

LE PROGRAMME

Lignes 20 à 50

On initialise, dans ces lignes, les différents paramètres du jeu (voir identification des variables).

Lignes 100 à 340

On définit la largeur du pied et la hauteur de la montagne par deux nombres aléatoires XM et YM, puis l'on dessine cette montagne en utilisant l'instruction **PLOT**.

Lignes 350 à 900

Pour entrer le mouvement et la quantité de carburant nécessaire, on utilise le sous-programme des lignes 2500 à 3210. On incrémente (ou l'on décrémente) la position de la fusée en fonction de ces valeurs.

L'appel du sous-programme en 4000 déplace la fusée sur l'écran en prenant les anciennes coordonnées pour la position de départ et les coordonnées nouvellement calculées pour sa position d'arrivée.

Après affichage de l'oxygène et du carburant restant, on teste s'ils sont supérieurs à zéro. Si ce n'est pas le cas, on affiche un message de condoléances correspondant au type d'erreur de vol commis (à partir de la ligne 5000).

Dans le cas où les nouvelles coordonnées de la fusée correspondent à celles de l'arrivée, on saute aux lignes 6000 pour le message de félicitations et le renouvellement du carburant et de l'oxygène.

Lignes 2000 à 2300

On efface, par l'instruction **UNPLOT**, la fusée à ses anciennes coordonnées et on la dessine à sa nouvelle position.

Dans le cas où l'on sort du cadre de l'écran, un des messages des lignes 5000 apparaît.

Lignes 4000 à 4100

La trajectoire de la fusée, suivant ses anciennes et nouvelles coordonnées, est calculée dans ces lignes. A chaque nouvel incrément de sa position, le sous-programme en 2000 la déplace.

DATE	NOM DU PROGRAMME : FUSÉE LUNAIRE	PAGE 1/2

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
APH	VARIABLE		X				Accélération verticale
APV	VARIABLE		X				et horizontale
AXF	VARIABLE		X				anciennes coordonnées de la
AYF	VARIABLE		X				fuseé avant déplacement
CA	COMPTEUR		X				Carburant restant
DXF	VARIABLE		X				Variable de Sauvegarde des
DYF	VARIABLE		X				coordonneés verticales et
							horizontales de la fuseé
							pendant le déplacement
I	INDICE		X				Boucle FOR / NEXT
INC	VARIABLE		X				Incrément du déplacement
							horizontal de la fusée
M$	VARIABLE		X				Message d'erreur
OX	COMPTEUR		X				oxygène restant
PH	VARIABLE		X				poussée horizontale
PV	VARIABLE		X				poussée verticale
Q	VARIABLE	X					Quantité de carburant à
							utiliser
S$	VARIABLE	X					Sens du déplacement
ST	VARIABLE		X				Incrément du déplacement
							vertical de la fusée.

6.1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

DATE	NOM DU PROGRAMME : FUSÉE LUNAIRE	PAGE 2/2

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
XF	VARIABLE		X				Coordonnées de la
YF	VARIABLE		X				fusée.
XM	VARIABLE		X				Coordonnées de la
YM	VARIABLE		X				montagne.

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Morpion vertical

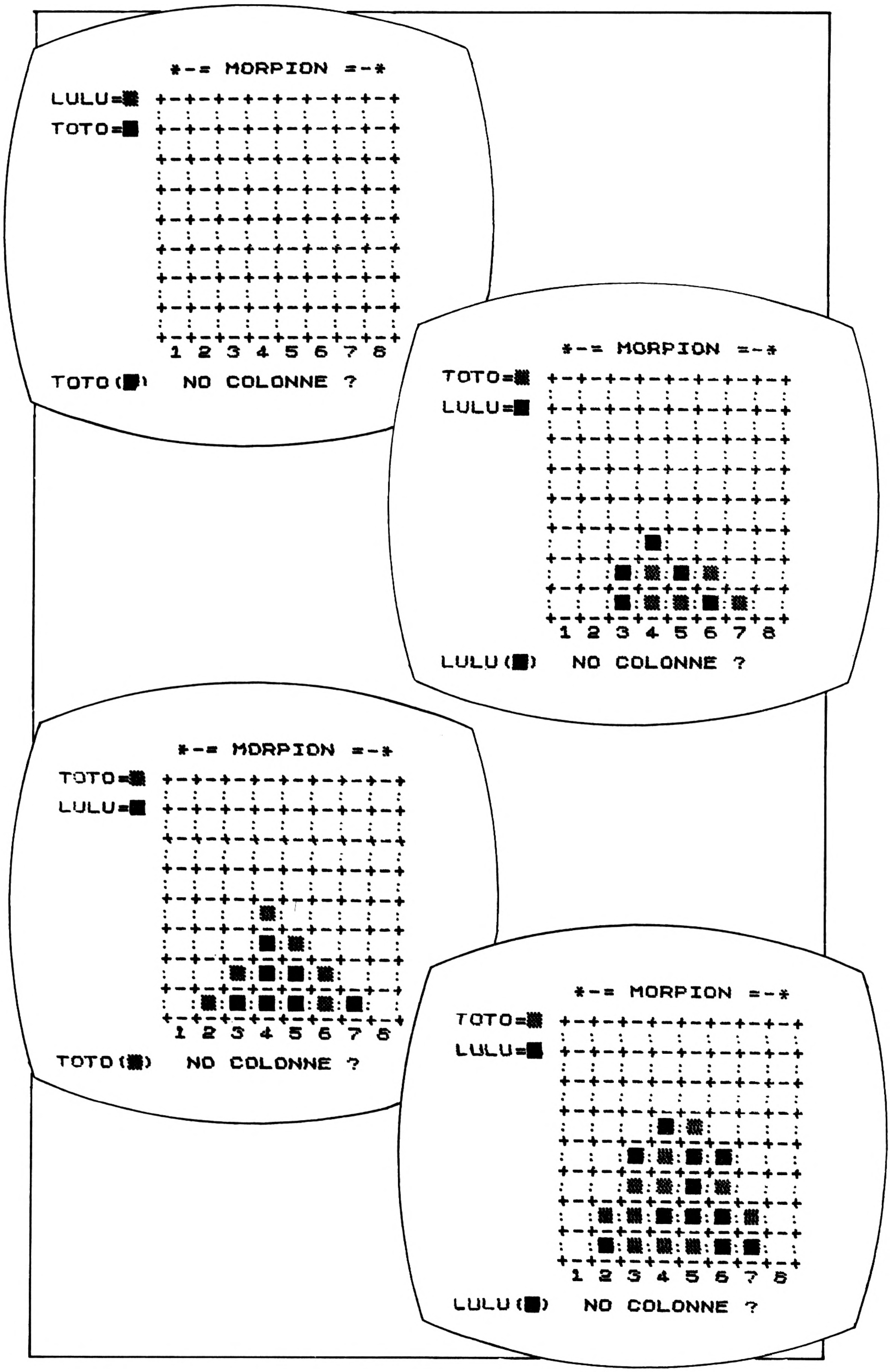
Vous connaissez certainement le jeu du morpion que le ZX81 vous permet, ici, de jouer à deux.

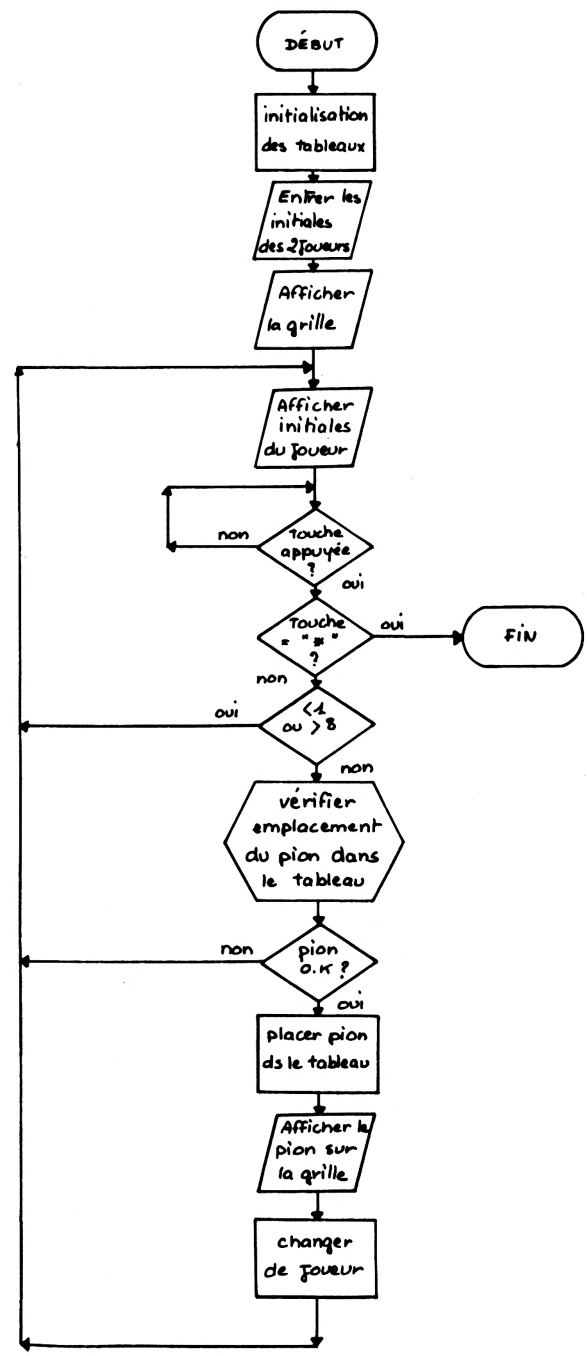
Il vous demande, d'abord, les initiales des deux joueurs (à donner en quatre lettres maximum).

Puis, il trace, sur l'écran, une grille de 8 x 8 cases et attend que le premier joueur, dont la couleur est signalée en haut à gauche de l'écran (carré gris ou noir), tape le numéro de la colonne où tombera son pion.

Puis, c'est le tour du deuxième joueur et ainsi de suite. Il suffit d'appuyer sur une des touches de 1 à 8, et le déplacement s'effectue.

Au gré des deux joueurs, celui qui gagne la partie peut être le premier qui aligne horizontalement, verticalement ou en diagonale, 4 ou 5 pions. C'est à eux d'en décider, car une partie ne s'arrête pas automatiquement, mais en rentrant "*" à la place d'un numéro. Une nouvelle grille se dessinera alors, et le jeu repartira.


-= MORPION =-
LULU=
TOTO=
1 2 3 4 5 6 7 8
TOTO( ) NO COLONNE ?
-= MORPION =-
TOTO=
LULU=
1 2 3 4 5 6 7 8
LULU( ) NO COLONNE ?
-= MORPION =-
TOTO=
LULU=
1 2 3 4 5 6 7 8
TOTO( ) NO COLONNE ?
-= MORPION =-
TOTO=
LULU=
1 2 3 4 5 6 7 8
LULU( ) NO COLONNE ?


DÉBUT
initialisation des tableaux
Entrer les initiales des 2 Joueurs
Afficher la grille
Afficher initiales du Joueur
non
Touche appuyée ?
oui
Touche = "#" ?
oui
FIN
non
oui
<1 ou >8
non
vérifier emplacement du pion dans le tableau
non
pion O.K ?
oui
placer pion ds le tableau
Afficher le pion sur la grille
changer de Joueur

```
  10 REM MORPION VERTICAL
  20 REM ------------------
  30 DIM P(10,10)
  40 DIM N$(2,4)
  45 DIM P$(2,1)
  50 FOR I=1 TO 2
  60 PRINT AT 21,0;"INITIALES DU
 JOUEUR ";I
  70 INPUT N$(I)
  80 NEXT I
  90 CLS
  95 REM ----------------------
  97 REM DESSIN DE LA GRILLE
 100 PRINT TAB 8;"*-= MORPION =-
*"
 105 PRINT
 110 FOR I=1 TO 9
 120 PRINT TAB 7;"+";
 130 FOR J=1 TO 8
 140 PRINT "-+";
 150 NEXT J
 160 PRINT
 165 IF I=9 THEN GOTO 230
 170 PRINT TAB 7;":";
 180 FOR J=1 TO 8
 190 PRINT " :";
 200 NEXT J
 210 PRINT
 220 NEXT I
 230 PRINT TAB 7;
 240 FOR I=1 TO 8
 250 PRINT " ";I;
 260 NEXT I
 270 LET P$(1)="▒"
 280 LET P$(2)="█"
 290 PRINT AT 2,0;N$(1);"=▒"
 300 PRINT AT 4,0;N$(2);"=█"
 310 LET D=INT (RND*2)+1
 380 REM ---------------------
 390 REM JEU
 400 PRINT AT 21,0;N$(D);"(";P$(
D);")   NO COLONNE ?"
 410 IF INKEY$="" THEN GOTO 410
 420 LET R$=INKEY$
 430 IF R$="*" THEN RUN
 440 LET R=VAL R$
 450 IF R<1 OR R>8 THEN GOTO 400
 460 FOR I=1 TO 8
 470 IF P(R,I)<>0 THEN GOTO 490
 480 NEXT I
 490 LET I=I-1
 500 IF I=0 THEN GOTO 400
 510 LET P(R,I)=D
 580 REM ---------------------
 590 REM AFFICHAGE DES PIONS
 600 FOR J=1 TO 8
 610 PRINT AT (J*2)+1,(R*2)+6;P$
(D)
 620 IF I=J THEN GOTO 700
 630 PRINT AT (J*2)+1,(R*2)+6;"
"
 640 NEXT J
 680 REM ---------------------
 690 REM CHANGEMENT DE JOUEUR
 700 IF D=1 THEN GOTO 730
 710 LET D=1
 720 GOTO 400
 730 LET D=2
 740 GOTO 400
```

LE PROGRAMME

Lignes 30 à 90

On dimensionne trois tableaux :

- **P** pour connaître l'emplacement des pions sur la grille.
- **N$** pour les initiales des joueurs.
- **P$** pour les caractères graphiques représentant les pions de chaque joueur.

On saisit, aux lignes 50 à 80, les initiales des deux joueurs en utilisant une **boucle FOR/NEXT.**

Lignes 100 à 310

On dessine, sur l'écran, la grille du jeu en utilisant plusieurs boucles imbriquées. Puis l'on assigne, au tableau P$, les caractères graphiques correspondant aux pions. On les affiche en haut à gauche de l'écran avec les initiales de chaque joueur.

En plaçant, dans la variable D, un nombre aléatoire compris entre 1 et 2, on désigne le joueur qui doit commencer la partie.

Lignes 400 à 510

Après affichage du nom du joueur, on saisit, au clavier, le numéro de la colonne choisie. Si une étoile est tapée, on considère le jeu comme terminé et l'on retourne au début par l'instruction **RUN.**

Pour la colonne désignée, on recherche, dans le tableau P, la première case libre. Si la colonne est déjà remplie, on retourne à la saisie (ligne 400), sinon on place dans le tableau le numéro du joueur.

Lignes 600 à 640

En partant du haut de la grille, on affiche le pion du joueur que l'on efface immédiatement tant que l'on n'est pas sur la première case inoccupée.

Lignes 700 à 740

Avant de retourner à la saisie d'une autre colonne, on place dans la variable D le numéro du joueur suivant.

DATE	NOM DU PROGRAMME : MORPION VERTICAL	PAGE 1

IDENTIFICATION des VARIABLES

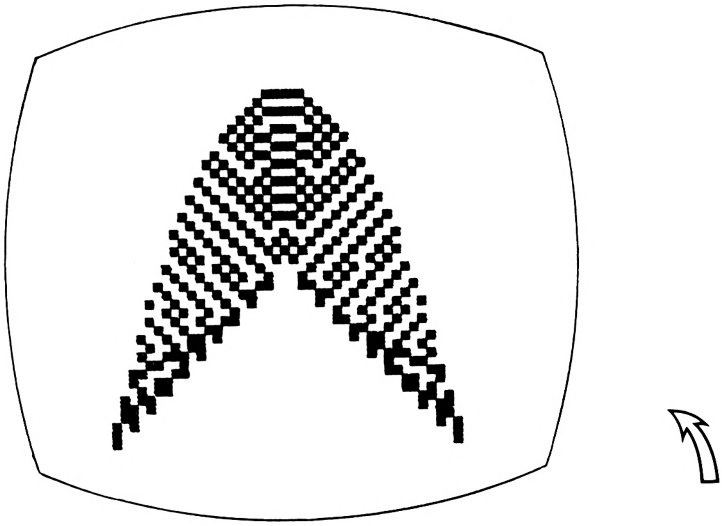
NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	Programme	Cassette	Disquette	LONG.	OBSERVATIONS
D	DRAPEAU						Numéro du Joueur en cours
I	INDICE						Boucle FOR / NEXT
J	INDICE						Boucle FOR / NEXT
N$( )	TABLEAU						Initiales des Joueurs
P( )	TABLEAU						Tableau contenant les pions
							de chaque Joueur.
P$( )	TABLEAU						Représentation graphique des
							pions de chaque Joueur
R	VARIABLE						Identique à R$ mais Valeur
							numérique
R$	VARIABLE						Réponse du Joueur
							(N° de colonne ou * pour
							fin de partie).

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

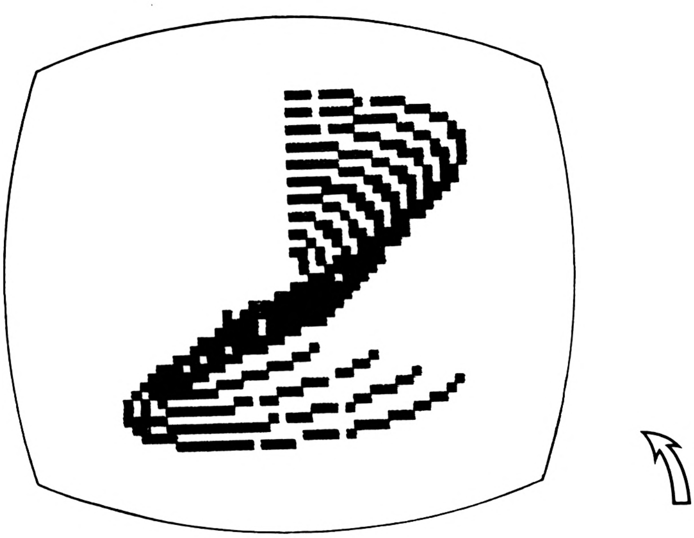
Graphiques

Les quatre programmes qui suivent ont quelque chose en commun. Ils sont en effet identiques, à l'exception d'une seule ligne. Mais quel changement dans les résultats ...!

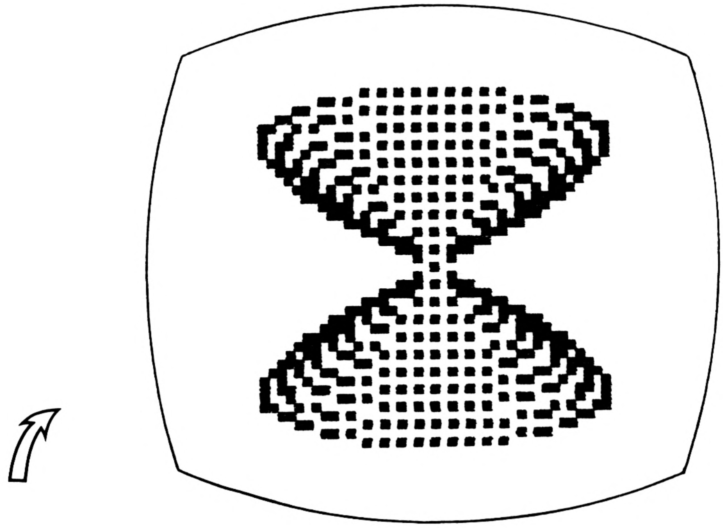
Alors ... donnez libre cours à votre imagination et modifiez la ligne 30 pour obtenir de nouveaux graphiques de plus en plus originaux.



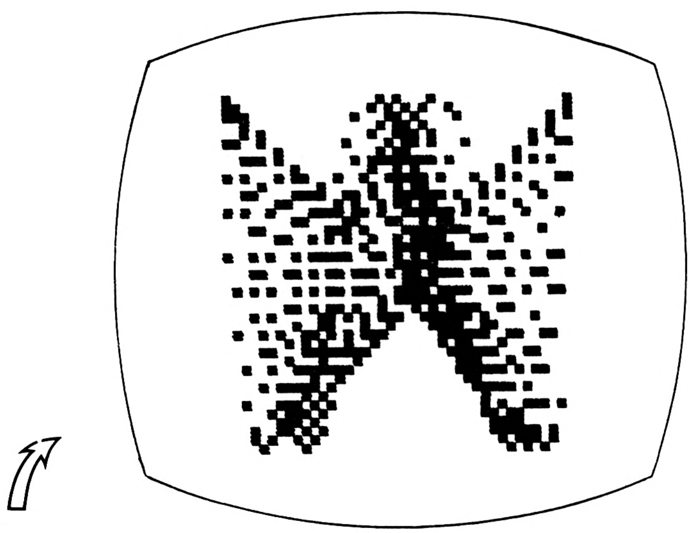
```
 10 FOR J=1 TO 10
 20 FOR I=0 TO J*12
 30 PLOT 32+J*2*SIN (I/(J*6)*PI
),22+J*2*COS (I/(J*3)*PI)
 40 NEXT I
 50 NEXT J
```



```
  1 FAST
 10 FOR J=1 TO 10
 20 FOR I=0 TO J*12
 30 PLOT 32+J*2*SIN (I/(J+6)),2
2+J*2*COS (I/(J*3))
 40 NEXT I
 50 NEXT J
 60 SLOW
```



```
   10 FOR J=1 TO 10
   20 FOR I=0 TO J*12
   30 PLOT 32+J*2*SIN (I/(J*3)*PI
),22+J*2*COS (I/(J*6)*PI)
   40 NEXT I
   50 NEXT J
```



```
    1 FAST
   10 FOR J=1 TO 10
   20 FOR I=0 TO J*12
   30 PLOT 32+J*2*SIN (I/(J+6)),2
2+J*2*COS (I/(J+3)*PI)
   40 NEXT I
   50 NEXT J
   60 SLOW
```

Tours de Hanoï

Au départ, plusieurs disques d'inégales largeurs sont empilés sur une tour située à gauche de l'écran. Il y a une deuxième tour vide au centre et une troisième tour vide à droite. Il s'agit de mettre les disques de la tour gauche sur la tour droite et dans le même ordre.

Dès le début, un message "**NOMBRE DE DISQUES(3 à 7) ?**" vous demande avec combien de disques vous voulez jouer.
Puis les tours se dessinent et le ZX81 vous demande votre mouvement par :

"TOUR DE DEPART" : à entrer 1, 2 ou 3 (numéro des tours de gauche à droite)

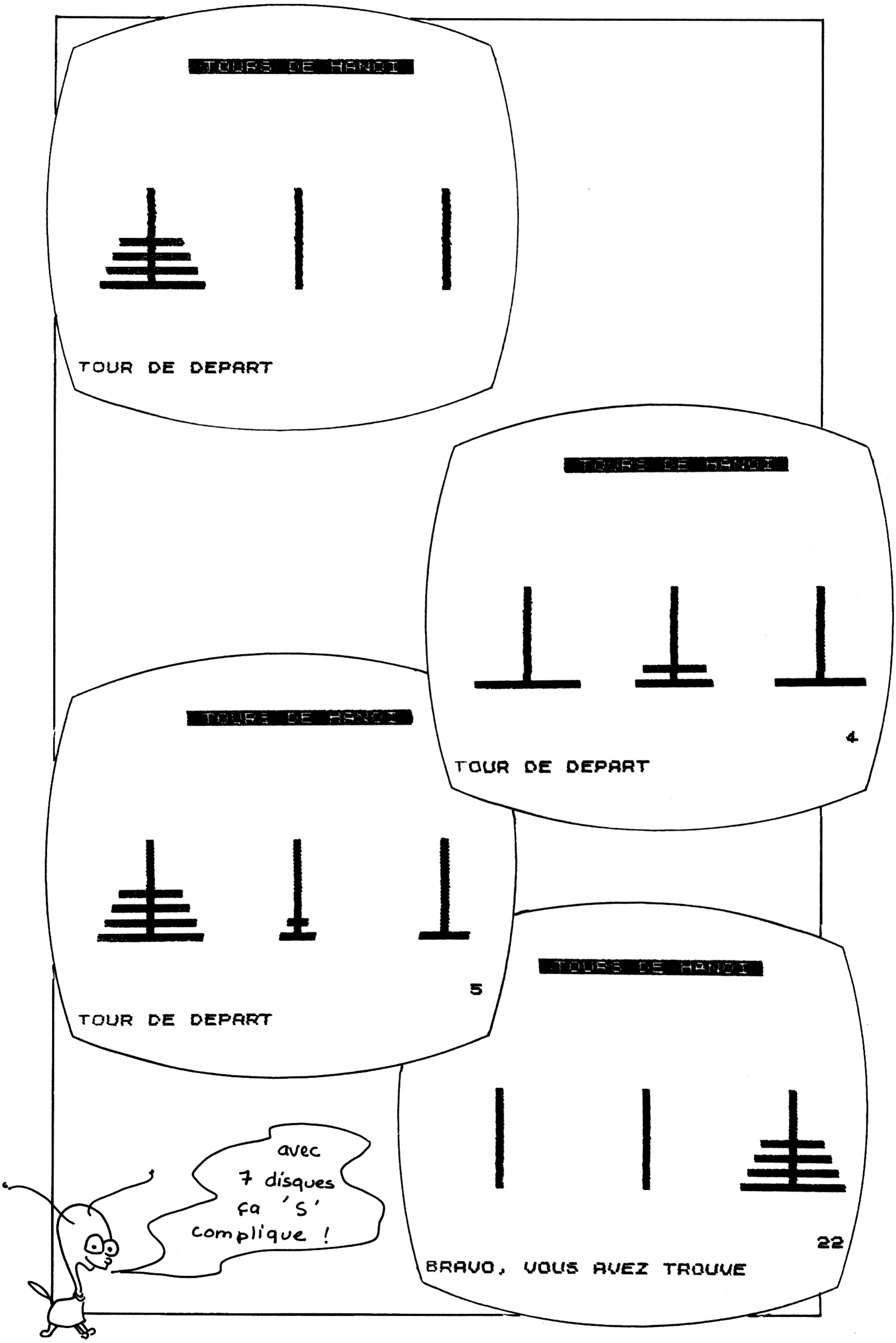
et **"TOUR D'ARRIVEE"** : à entrer,

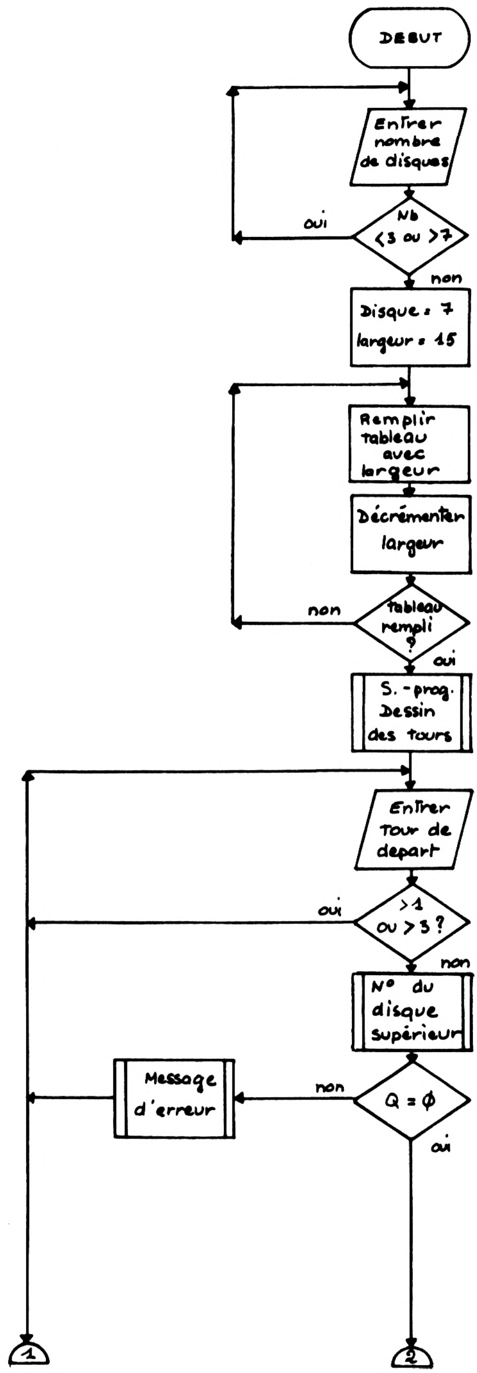
jusqu'à ce que tous les disques soient sur la troisième tour.

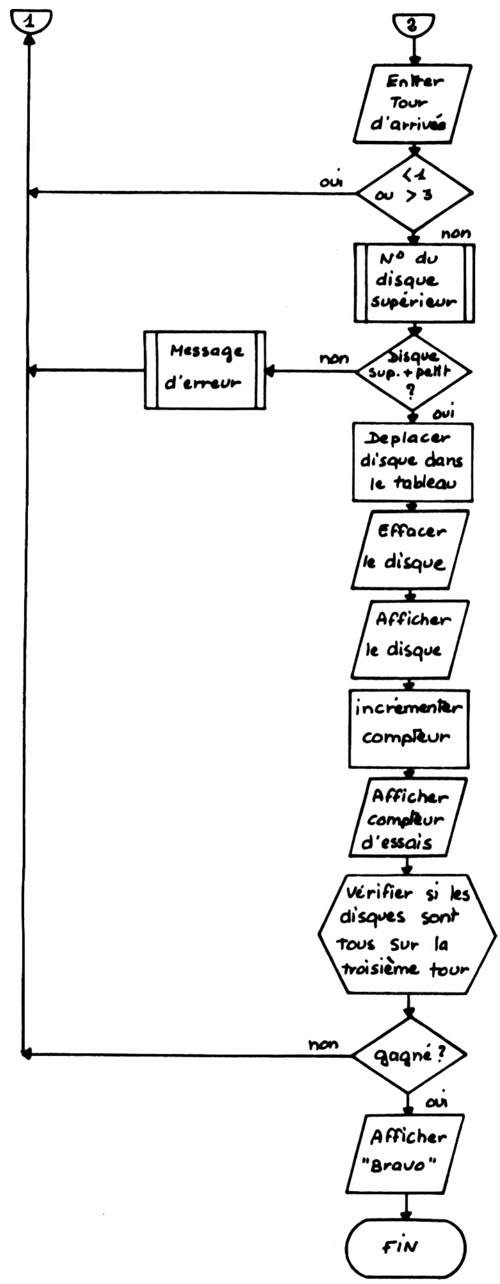
Le message **"BRAVO, VOUS AVEZ TROUVE"** récompense alors vos efforts.

Au cours de jeu, en cas de fausse manoeuvre, un message d'erreur s'affichera. Il est, en effet, impossible de poser un disque sur un autre plus petit. Par contre, un disque peut aller de la tour 1 à la tour 3, puis revenir à la tour 2...

Si vous voulez un autre jeu, il suffit de taper sur n'importe quelle touche à l'apparition du message **"BRAVO"**.


TOURS DE HANOI
TOUR DE DEPART
TOURS DE HANOI
4
TOUR DE DEPART
TOURS DE HANOI
5
TOUR DE DEPART
TOURS DE HANOI
22
BRAVO, VOUS AVEZ TROUVE
avec 7 disques ça 'S' complique !


DEBUT
Entrer nombre de disques
Nb <3 ou >7
oui
non
Disque = 7
largeur = 15
Remplir tableau avec largeur
Décrémenter largeur
tableau rempli ?
non
oui
S.-prog. Dessin des tours
Entrer Tour de depart
>1 ou >3 ?
oui
non
N° du disque supérieur
Q = ∅
non
oui
Message d'erreur
1
2


1
2
Entrer
Tour
d'arrivée
<1
ou >3
oui
non
N° du
disque
supérieur
Disque
sup. + petit
?
non
oui
Message
d'erreur
Deplacer
disque dans
le tableau
Effacer
le disque
Afficher
le disque
incrementer
compteur
Afficher
compteur
d'essais
Vérifier si les
disques sont
tous sur la
troisième tour
gagné ?
non
oui
Afficher
"Bravo"
FIN

```
  10 REM TOURS DE HANOI
  20 REM ------------------------
  30 DIM T(7,3)
  40 LET C=0
 100 PRINT AT 21,0;"NB DE DISQUE
S (3 A 7)  ?"
 110 INPUT NBD
 120 IF NBD<3 OR NBD>7 THEN GOTO
 100
 130 CLS
 140 PRINT TAB 8;"TOURS DE HANO
I"
 200 LET D=7
 210 LET L=15 30
 220 FOR X=NBD TO 1 STEP -1
 230 LET T(D,1)=L
 240 LET L=L-2
 250 LET D=D-1
 260 NEXT X
 270 GOSUB 2000
 290 REM --------------------
 300 REM JEU
 310 PRINT AT 21,0;"TOUR DE DEPA
RT"
 320 INPUT TD
 323 IF TD<1 OR TD>3 THEN GOTO 3
10
 330 LET NT=TD
 340 GOSUB 3000
 350 IF Q=0 THEN GOTO 390
 360 LET M$="PAS DE DISQUE"
 370 GOSUB 4000
 380 GOTO 310
 390 LET DD=I
 400 PRINT AT 21,0;"TOUR D ARRIV
EE"
 410 INPUT TA
 420 IF TA<1 OR TA>3 OR TA=TD TH
EN GOTO 310
 430 LET NT=TA
 440 GOSUB 3000
 450 LET DA=I-1
 460 IF DA=7 THEN GOTO 500
 465 IF T(DD,TD)<T(DA+1,TA) THEN
 GOTO 500
 470 LET M$="DISQUE SUP. PLUS GR
AND"
 480 GOSUB 4000
 490 GOTO 310
 500 LET T(DA,TA)=T(DD,TD)
 510 LET T(DD,TD)=0
 520 FOR K=1 TO 15
 530 UNPLOT K+(21*(TD-1))+2,((8-
DD)*2)+10
 540 NEXT K
 550 PLOT (21*(TD-1))+10,((8-DD)
*2)+10
 560 FOR K=1 TO T(DA,TA)
 570 PLOT ((15-T(DA,TA))/2)+2+(2
1*(TA-1))+K,((8-DA)*2)+10
 580 NEXT K
 590 LET C=C+1
 600 PRINT AT 19,28;C
 610 FOR I=8-NBD TO 7
 620 IF T(I,3)=0 THEN GOTO 310
 630 NEXT I
 640 PRINT AT 21,0;"BRAVO, VOUS
AVEZ TROUVE"
 650 IF INKEY$="" THEN GOTO 650
 660 RUN
```

```
1980 REM -------------------
1990 REM DESSIN DES TOURS
2000 REM
2010 FOR J=1 TO 7
2020 IF T(J,1)=0 THEN GOTO 2090
2030 LET X=((15-T(J,1))/2)+2
2040 FOR K=1 TO T(J,1)
2050 PLOT X+K,((8-J)*2)+10
2060 NEXT K
2090 NEXT J
2130 FOR I=1 TO 3
2140 FOR K=12 TO 25
2150 PLOT 10+(21*(I-1)),K
2160 NEXT K
2170 NEXT I
2180 RETURN
2980 REM -------------------
2990 REM NO DU DISQUE SUP.
3000 FOR I=1 TO 7
3010 IF T(I,NT)<>0 THEN GOTO 310
0
3020 NEXT I
3070 LET Q=1
3080 RETURN
3100 LET Q=0
3110 RETURN
3980 REM -------------------
3990 REM MESSAGE D ERREUR
4000 FOR I=1 TO 25
4010 PRINT AT 20,0;M$
4020 PRINT AT 20,0;"
"
4030 NEXT I
4040 RETURN
```

LE PROGRAMME

Lignes 30 à 270

On dimensionne un tableau T de sept sur trois, pour connaître, pendant le jeu, l'emplacement des sept disques sur les trois tours.

Puis, l'on saisit le nombre de disques à jouer entre trois et sept.

On remplit le tableau de départ en mettant tous les disques sur la première tour, le disque le plus grand étant placé à la position 7.

Le sous-programme des lignes 2000 à 2180 dessine le jeu du début, c'est-à-dire la première tour avec le nombre de disques que vous avez choisi et les autres tours vides.

Lignes 31Ø à 49Ø

On commence le jeu en saisissant le numéro de la tour de départ (on déplace, dans tous les cas, le disque supérieur). Le sous-programme en 3000 trouve le numéro de ce disque. Si cette tour ne comporte aucun disque, on met le drapeau à 1 (variable Q), et le sous-programme en 4000 affiche le message clignotant *"PAS DE DISQUE"* ; puis, l'on revient à la saisie (ligne 310).

On utilise le même principe pour la tour d'arrivée. Au retour du sous-programme en 3000, on teste si le disque du dessus n'est pas plus petit. Si c'est le cas, le sous-programme en 4000 affiche le message clignotant *"DISQUE SUPERIEUR PLUS GRAND"*, et l'on revient à la saisie de la tour de départ (ligne 310).

Lignes 5ØØ à 6ØØ

On déplace le disque de la tour de départ sur la tour d'arrivée dans le tableau T. Comme on efface une partie de la tour de départ, la ligne 550 replace la partie manquante.

La ligne 590 incrémente le compteur d'essais, puis la ligne 600 l'affiche sur la ligne 19 de l'écran.

Lignes 61Ø à 66Ø

On teste, sur la troisième tour, le nombre de disques déjà placés. S'il correspond au nombre total de disques, on affiche le message *"BRAVO"*, puis l'on attend qu'une touche soit appuyée pour commencer un autre jeu. Dans le cas contraire, on retourne à la ligne 310.

DATE	NOM DU PROGRAMME : TOURS DE HANOÏ	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
C	COMPTEUR		X				Nombre d'essais
D	VARIABLE		X				Nombre maximum de disques
DA	VARIABLE		X				Emplacement du disque dans
							la tour d'arrivée
DD	VARIABLE		X				Emplacement du disque dans
							la tour de départ
I	INDICE		X				Boucle FOR / NEXT
J	INDICE		X				Boucle FOR / NEXT
K	INDICE		X				Boucle FOR / NEXT
L	VARIABLE		X				Largeur du grand disque
M$	VARIABLE		X				Message d'erreur
NBD	VARIABLE	X					Nombre de disques
NT	VARIABLE		X				Numéro de la tour à étudier
							dans le sous-programme 3000
Q	DRAPEAU		X				Si Q = 1 → erreur de saisie
T	TABLEAU		X				Tableau contenant les positions
							des disques de chaque tour.
TA	VARIABLE	X					Tour d'arrivée
TD	VARIABLE	X					Tour de départ
X	INDICE		X				Boucle FOR / NEXT.

6/1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Alphabet

Quel est l'heureux parent qui ne désire pas savourer un quart d'heure de sérénité, tout en sachant que les chers bambins ne sont pas en train de démonter le lavabo... Voici, pour eux, un programme qui joindra l'utile à l'agréable.

Dès le début, l'alphabet se dessine sur l'écran, sauf une lettre remplacée par un carré noir et désignée par une flèche.

On peut essayer jusqu'à trois lettres, en réponse de la lettre manquante. Si le ZX81 affiche un carré noir, c'est que la réponse est bonne.

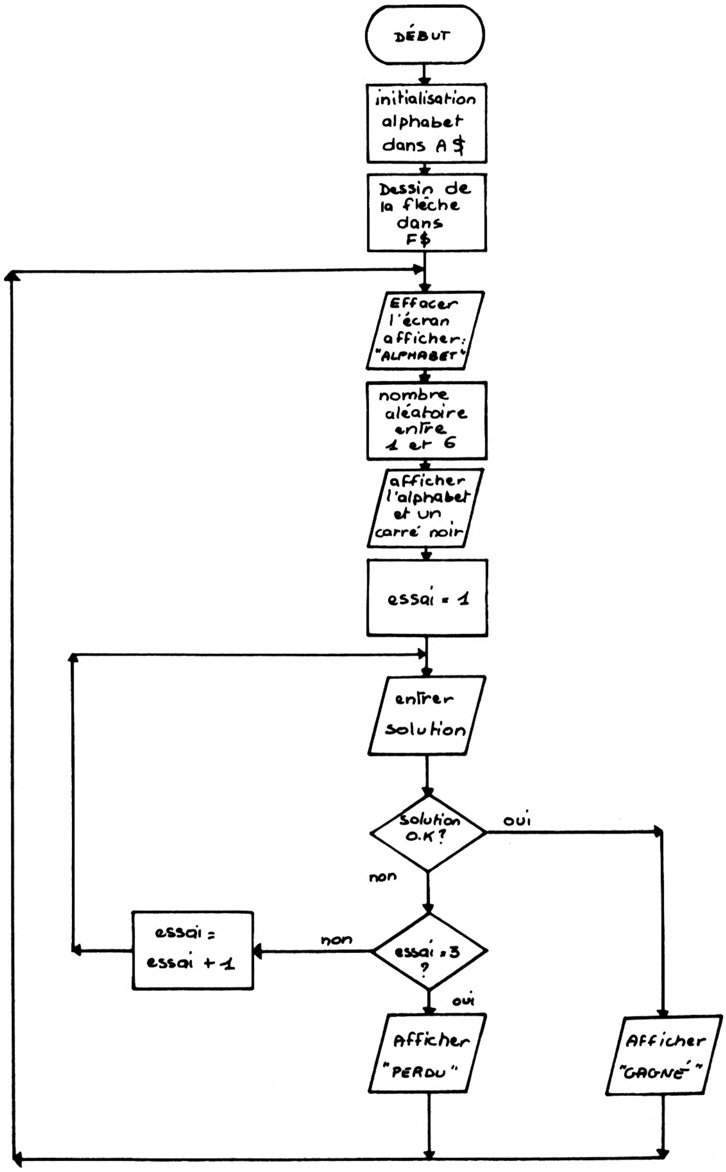
A la première bonne réponse, le message "carré noir -GAGNE- carré noir" s'inscrit en clignotant ; le carré noir étant un rappel du carré noir de la lettre à trouver.

Si toutes les réponses sont fausses, le message "carré gris -PERDU- carré gris" s'inscrit en clignotant ; le carré gris signifiant une erreur.

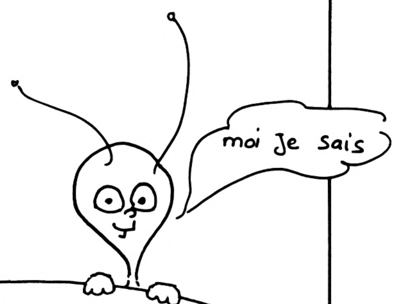
Puis, le jeu repart tout seul au bout de quelques instants.

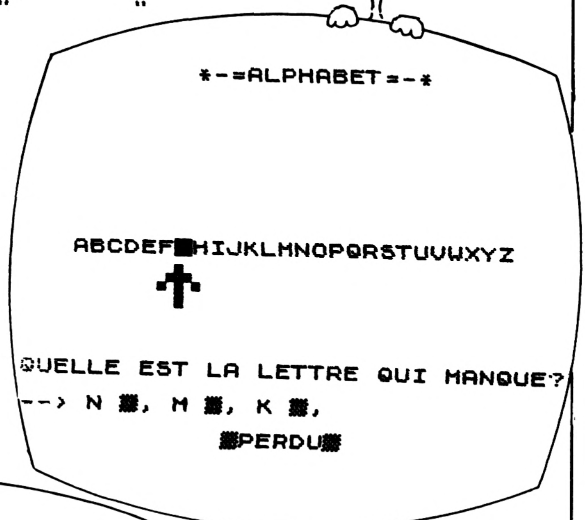
Les carrés noirs et gris sont les moyens visuels pour les enfants qui jouent seuls (et qui ne savent pas forcément lire) de savoir si la lettre qu'ils donnent est bonne ou mauvaise.

Le manque de graphisme précis du ZX81 empêche des messages plus parlants, tel un bonhomme à l'air heureux ou malheureux. Mais les programmeurs persévérants pourront essayer de trouver des messages tout aussi suggestifs !


DÉBUT
initialisation alphabet dans A $
Dessin de la flèche dans F$
Effacer l'écran afficher: "ALPHABET"
nombre aléatoire entre 1 et 6
afficher l'alphabet et un carré noir
essai = 1
entrer solution
Solution O.K ?
oui
non
essai = 3 ?
non
essai = essai + 1
oui
Afficher "PERDU"
Afficher "GAGNÉ"

```
 10 REM ALPHABET
 20 REM ---------------------
 30 LET A$="ABCDEFGHIJKLMNOPQRS
TUVWXYZ"
 40 DIM F$(3,3)
 50 LET F$(1)=" ▄ "
 60 LET F$(2)="▄█▄"
 70 LET F$(3)=" █ "
 90 CLS
100 PRINT TAB 10;"*-=ALPHABET=-
*"
110 LET X=INT (RND*26)+1
120 PRINT AT 10,3;A$
130 PRINT AT 10,2+X;"■"
140 FOR I=1 TO 3
150 PRINT AT 10+I,1+X;F$(I)
160 NEXT I
170 PRINT AT 17,0;"QUELLE EST L
A LETTRE QUI MANQUE?"
175 PRINT AT 19,0;"-->";
180 FOR J=1 TO 3
190 IF INKEY$="" THEN GOTO 190
200 LET R$=INKEY$
210 PRINT " ";R$;
220 IF R$=CHR$ (37+X) THEN GOTO
300
230 PRINT " ▒,";
240 LET R$=INKEY$
250 NEXT J
260 LET G$="▒PERDU▒"
270 GOTO 400
300 PRINT " █"
310 LET G$="█GAGNE█"
400 FOR K=1 TO 80
410 PRINT AT 21,12;G$
420 PRINT AT 21,12;"       "
430 NEXT K
440 GOTO 90
```







LE PROGRAMME

Lignes 30 à 70

On initialise la variable A$ avec l'alphabet, ce qui est plus rapide que de passer par une boucle et d'afficher, à la suite les uns des autres, les codes du ZX81 de 38 à 63 (A à Z).

Après dimensionnement, on place, dans la variable F$, le dessin d'une flèche.

Lignes 90 à 340

Après effacement de l'écran (**CLS**), on affiche le titre "ALPHABET", puis l'on place, dans X, un nombre aléatoire entre 1 et 26, pour désigner le caractère de l'alphabet que l'on doit trouver.

En lignes 120 et 130, on affiche l'alphabet et un carré noir sur le caractère pointé par X.

Acceptant trois réponses, on passe par une boucle **FOR NEXT**. Si la réponse correspond au caractère à trouver, on saute à la ligne 300 qui place dans la variable G$ le message "GAGNE". Si on sort de la boucle normalement (par le NEXT J), on place le message "PERDU" dans G$.

Lignes 400 à 440

On affiche successivement (et quatre vingt fois) le message G$ que l'on efface immédiatement pour donner un effet de clignotement.

DATE	NOM DU PROGRAMME : ALPHABET	PAGE /

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	Clavier	Programme	Cassette	Disquette	LONG.	OBSERVATIONS
A$	VARIABLE		X				Chaîne de caractères contenant
							l'alphabet
F$( )	TABLEAU		X				Dessin d'une flèche
G$	VARIABLE		X				Contient "GAGNE" ou
							"PERDU"
I	INDICE		X				Boucle FOR / NEXT pour le
							dessin de la flèche
J	INDICE		X				Boucle FOR / NEXT des 3
							essais
K	INDICE		X				Boucle FOR/NEXT pour l'affichage
							du message G$
R$	VARIABLE	X					réponse du Joueur
X	VARIABLE		X				nombre aléatoire désignant
							la lettre à trouver

6.1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Tricot

Indispensable à toute tricoteuse, ce petit programme permet de connaître rapidement, sans calcul, le nombre de mailles ou de rangs à monter et à tricoter, et d'adapter ainsi, à toutes les tailles, les explications des tricots aux mesures standardisées.

Il suffit de rentrer le nombre des mailles et des rangs d'un échantillon tricoté de 10 cm sur 10 cm, les mesures diverses du dos et du devant (largeur à la base , largeur après diminutions des emmanchures, hauteur sous les bras, hauteur totale) et les mesures des manches (largeur poignet, largeur à l'épaule, hauteur à l'épaule, hauteur totale). Cela est demandé sur l'écran par l'affichage d'une page pour l'échantillon puis, automatiquement, par le passage à l'autre page des mesures. Au bout de quelques instants, une troisième page se dessine, donnant les mailles à monter, le nombre de mailles des diminutions ou des augmentations, ainsi que les rangs à tricoter, aussi bien pour le dos et le devant que pour les manches.

Entre chaque page, l'instruction **COPY**, intégrée au programme, lancera l'édition des données de l'écran. Au cas où vous ne possédez pas d'imprimante, il faudra supprimer les lignes 116, 475 et 980 et augmenter la valeur de **I** aux lignes 110 et 460 (boucle d'attente plus longue) pour avoir le temps de recopier l'écran.

La dernière page restera, dans ce cas, le temps que vous voudrez sur l'écran.




```
          ECHANTILLON
     10 CMS SUR 10 CMS


NOMBRE DE MAILLES ?
26 MAILLES


NOMBRE DE RANGS ?
34 RANGS
```

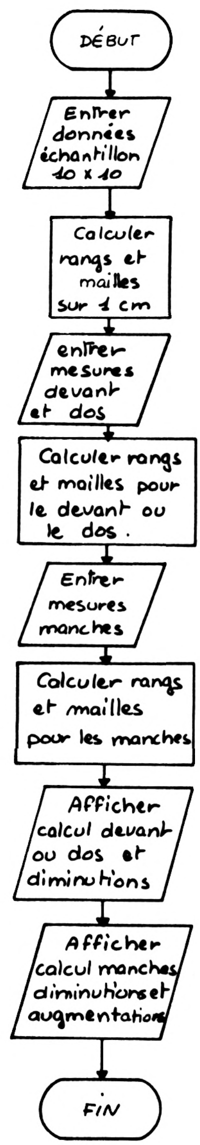
```
      MESURES DOS OU DEVANT
LARGEUR DOS DEVANT A LA BASE ?
48 CMS
LARGEUR POITRINE ?
40 CMS
HAUTEUR SOUS LES BRAS ?
40 CMS
HAUTEUR TOTALE DOS OU DEVANT ?
58 CMS

          MESURES MANCHES
LARGEUR POIGNET ?
22 CMS
LARGEUR MANCHE A L EPAULE ?
36 CMS
HAUTEUR MANCHE A L EPAULE ?
43 CMS
HAUTEUR TOTALE MANCHE ?
56 CMS
```

```
      MONTAGE DOS OU DEVANT
-MONTER 124 MAILLES,SOIT 48 CMS
-TRICOTER 136 RANGS,SOIT 40 CMS
-A 40 CMS, RABATTRE DE  CHAQUE
COTE 10 MAILLES, EN PLUSIEURS
FOIS, POUR LES EMMANCHURES.
-TRICOTER ENCORE 61 RANGS,SOIT
18 CMS.
-A 58 CMS,FERMER SELON L ENCO-
LURE.

          MONTAGE MANCHES
-MONTER 57 MAILLES, SOIT 22 CMS.
-TRICOTER 146 RANGS,SOIT 43 CMS,
EN AUGMENTANT REGULIEREMENT CHA-
QUE COTE DE 18 MAILLES.
-A 43 CMS,RABATTRE LES 93 MAIL-
LES RESTANTES REGULIEREMENT SOIT
 46 MAILLES DE CHAQUE COTE.
-A 56 CMS, FERMER LE TRICOT.
```




DÉBUT
Entrer données échantillon 10 x 10
Calculer rangs et mailles sur 1 cm
entrer mesures devant et dos
Calculer rangs et mailles pour le devant ou le dos.
Entrer mesures manches
Calculer rangs et mailles pour les manches
Afficher calcul devant ou dos et diminutions
Afficher calcul manches diminutions et augmentations
FIN

```
  10 REM MESURES TRICOT
  20 REM ----------------------
  30 REM ECHANTILLON 10 CMS SUR
10CMS
  40 PRINT AT 3,10;"ECHANTILLON"
  45 PRINT AT 4,7;"10 CMS SUR 10
 CMS"
  50 PRINT AT 9,0;"NOMBRE DE MAI
LLES ?"
  60 INPUT NMA
  65 PRINT AT 10,0;NMA;" MAILLES
"
  70 PRINT AT 15,0;"NOMBRE DE RA
NGS ?"
  80 INPUT NRA
  85 PRINT AT 16,0;NRA;" RANGS"
  90 LET MA=NMA/10
 100 LET RA=NRA/10
 110 FOR I=1 TO 50
 115 NEXT I
 116 COPY
 120 CLS
 130 REM ----------------------
 140 REM MESURES DEVANT ET DOS
 145 PRINT TAB 5;"MESURES DOS OU
DEVANT"
 150 PRINT "LARGEUR DOS DEVANT A
LA BASE ?"
 160 INPUT LDD
 170 PRINT LDD;" CMS"
 175 PRINT "LARGEUR POITRINE ?"
 176 INPUT LRP
 177 PRINT LRP;" CMS"
 180 PRINT "HAUTEUR SOUS LES BRA
S ?"
 190 INPUT HSB
 200 PRINT HSB;" CMS"
 210 PRINT "HAUTEUR TOTALE DOS O
U DEVANT ?"
 220 INPUT HDD
 230 PRINT HDD;" CMS"
 240 LET MLDD=INT (MA*LDD)
 250 LET RHSB=INT (RA*HSB)
 260 LET RHDD=INT (RA*HDD)
 265 LET MLRP=INT (MA*LRP)
 266 LET D=INT ((MLDD-MLRP)/2)
 270 PRINT
 275 PRINT
 280 REM ---------------------
 290 REM MESURES MANCHES
 295 PRINT TAB 7;"MESURES MANCHE
S"
 300 PRINT "LARGEUR POIGNET ?"
 310 INPUT LP
 320 PRINT LP;" CMS"
 330 PRINT "LARGEUR MANCHE A L E
PAULE ?"
 340 INPUT LE
 350 PRINT LE;" CMS"
 360 PRINT "HAUTEUR MANCHE A L E
PAULE ?"
 370 INPUT HE
 380 PRINT HE;" CMS"
 390 PRINT "HAUTEUR TOTALE MANCH
E ?"
 400 INPUT HT
 410 PRINT HT;" CMS"
 420 LET MLP=INT (MA*LP)
 430 LET MLE=INT (MA*LE)
 440 LET RHE=INT (RA*HE)
 450 LET RHT=INT (RA*HT)
 455 LET A=INT ((MLE-MLP)/2)
 460 FOR I=1 TO 50
 470 NEXT I
 475 COPY
 480 CLS
```

```
 700 REM ----------------------
 710 REM CALCULS DOS ET DEVANT
 720 PRINT TAB 5;"MONTAGE DOS OU
 DEVANT"
 730 PRINT "-MONTER ";MLDD;" MAI
LLES,SOIT ";LDD;" CMS."
 740 PRINT "-TRICOTER ";RHSB;" R
ANGS,SOIT ";HSB;" CMS."
 750 PRINT "-A ";HSB;" CMS, RABA
TTRE DE  CHAQUE  COTE ";D;" MAIL
LES, EN PLUSIEURS   FOIS, POUR L
ES EMMANCHURES."
 760 LET M=RHDD-RHSB
 770 LET N=HDD-HSB
 780 PRINT "-TRICOTER ENCORE ";M
;" RANGS,SOIT  ";N;" CMS."
 790 PRINT "-A ";HDD;" CMS,FERME
R SELON L ENCO-  LURE."
 800 PRINT
 810 PRINT
 900 REM -----------------------
 910 REM CALCULS MANCHES
 920 PRINT TAB 10;"MONTAGE MANCH
ES"
 930 PRINT "-MONTER ";MLP;" MAIL
LES, SOIT ";LP;" CMS."
 940 PRINT "-TRICOTER ";RHE;" RA
NGS,SOIT ";HE;" CMS,EN AUGMENTAN
T REGULIEREMENT CHA-QUE COTE DE 
";A;" MAILLES."
 950 LET O=(MLP+(A*2))
 955 LET P=INT (O/2)
 960 PRINT "-A ";HE;" CMS,RABATT
RE LES ";O;" MAIL- LES RESTANTES
 REGULIEREMENT,SOIT ";P;" MAILLE
S DE CHAQUE COTE."
 970 PRINT "-A ";HT;" CMS, FERME
R LE TRICOT."
 980 COPY
```

DATE	NOM DU PROGRAMME : TRICOT	PAGE 1/2

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
NMA	VARIABLE	X					Nb de mailles pour 10 cms
NRA	VARIABLE	X					Nb de rangs pour 10 cms
MA	VARIABLE		X				Nb de mailles pour 1 cm
RA	VARIABLE		X				Nb de rangs pour 1 cm
LDD	VARIABLE	X					largeur dos ou devant
LRP	VARIABLE	X					largeur à la poitrine
HSB	VARIABLE	X					Hauteur sous le bras
HDD	VARIABLE	X					Hauteur totale dos ou devant
MLDD	VARIABLE		X				Nb de mailles largeur dos ou devant
MLRP	VARIABLE		X				Nb de mailles largeur poitrine
RHSD	VARIABLE		X				Nb rangs hauteur sous-bras
RHDD	VARIABLE		X				Nb rangs hauteur totale dos ou devant
D	VARIABLE		X				Diminutions emmanchures de chaque
							coté
LP	VARIABLE	X					largeur poignet
LE	VARIABLE	X					largeur à l'epaule
HE	VARIABLE	X					Hauteur épaule
HT	VARIABLE	X					Hauteur Totale
MLP	VARIABLE		X				Nb de mailles poignet
MLE	VARIABLE		X				Nb de mailles à l'épaule
RHE	VARIABLE		X				Nb de rangs à l'epaule
RHT	VARIABLE		X				Nb de rangs hauteur totale

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

DATE	NOM DU PROGRAMME : TRICOT	PAGE 2/2

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	Programme	Cassette	Disquette	LONG.	OBSERVATIONS
A	VARIABLE		X				Augmentation manche de chaque
							côté
M	VARIABLE		X				Nombre de rangs entre l'épaule
							et le total dos ou devant.
N	VARIABLE		X				Cms entre l'épaule et le total
							dos ou devant.
O	VARIABLE		X				Nombre de mailles à l'épaule
							de la manche
P	VARIABLE		X				Nombre de mailles à rabattre de
							chaque côté de la manche.
I	INDICE		X				Boucle FOR / NEXT

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Ravitaillement en mer

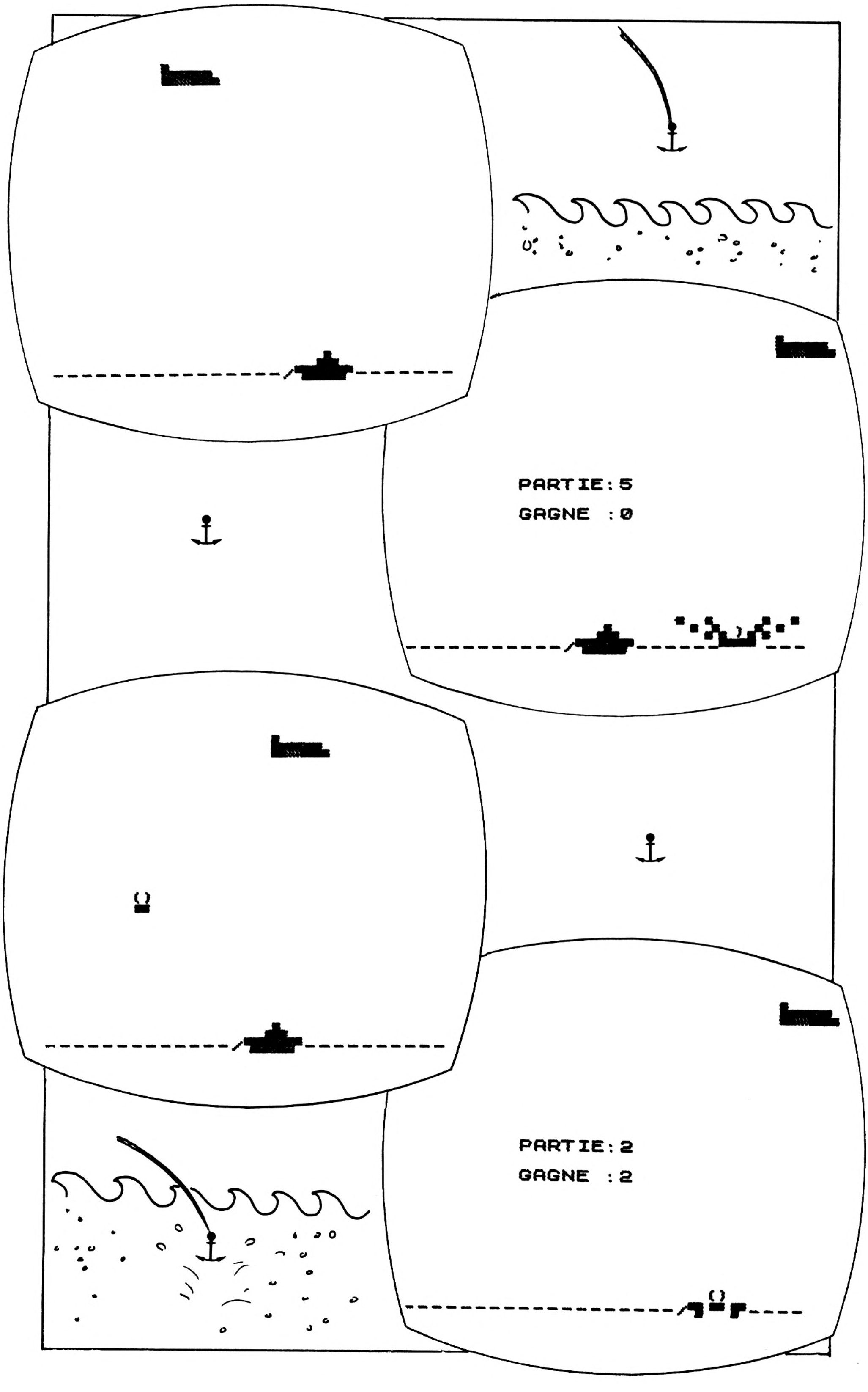
Un supertanker attend son ravitaillement que vous allez lui envoyer de votre avion.

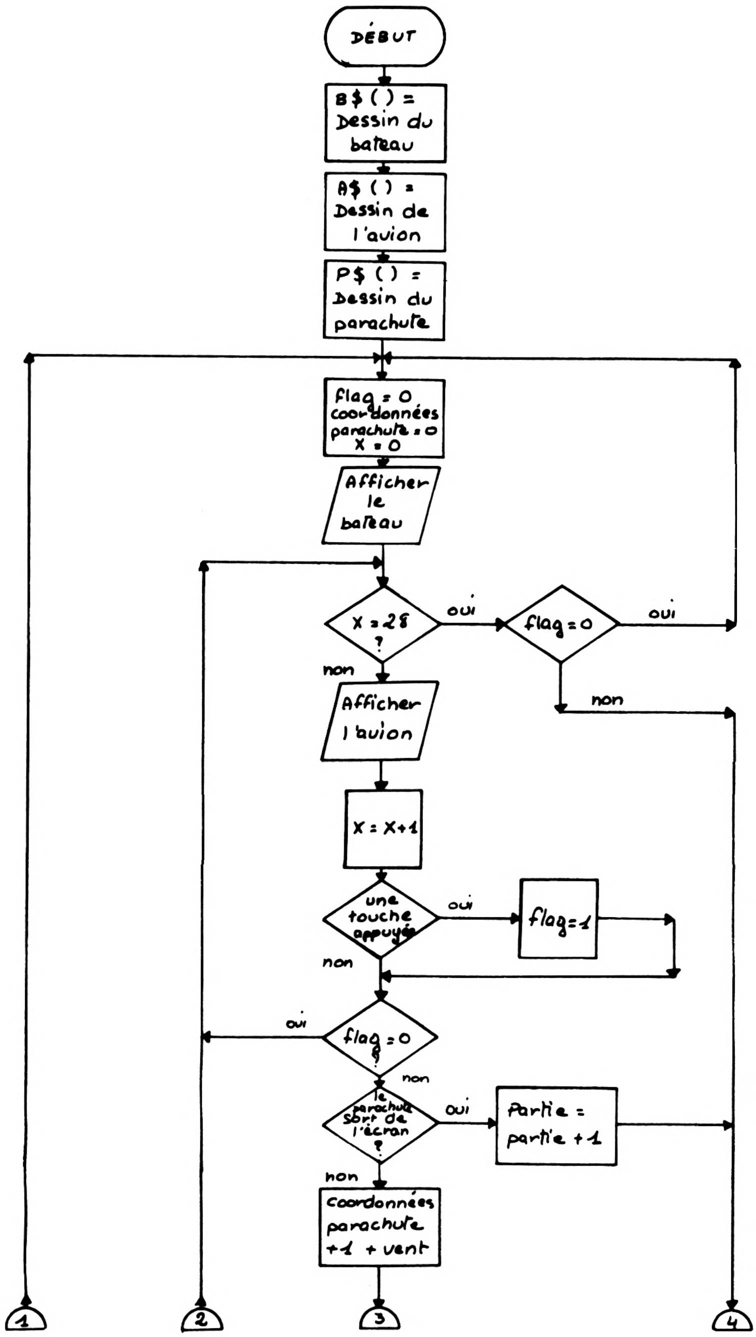
En tapant sur n'importe quelle touche (sauf BREAK), un parachute descendra les précieux colis.

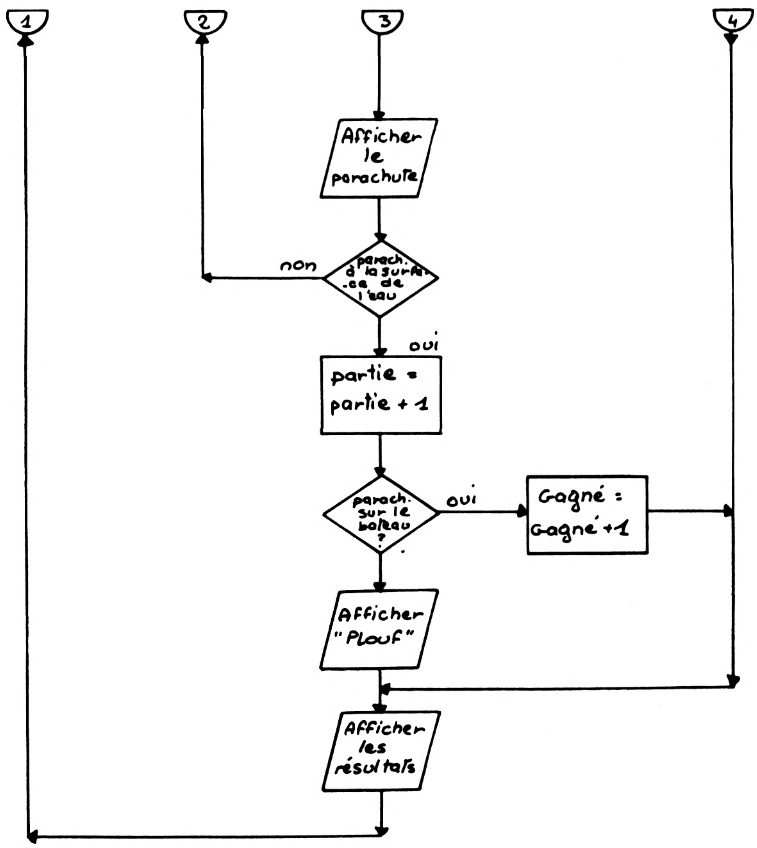
Mais attention, il peut y avoir du vent. Cela sera indiqué à gauche sur l'écran : à vous de prévoir la trajectoire corrigée.

Si vous n'êtes pas un bon viseur, adieu les colis !...

A chaque lancée, il y a un affichage des résultats, puis le jeu repart tout seul.


PARTIE:5
GAGNE :0
PARTIE:2
GAGNE :2


DÉBUT
B$ ( ) = Dessin du bateau
A$ ( ) = Dessin de l'avion
P$ ( ) = Dessin du parachute
flag = 0 coordonnées parachute = 0 X = 0
Afficher le bateau
X = 28 ?
oui
flag = 0
oui
non
non
Afficher l'avion
X = X+1
une touche appuyée
oui
flag = 1
non
oui
flag = 0 ?
non
le parachute sort de l'écran ?
oui
Partie = partie + 1
non
Coordonnées parachute +1 + vent
1
2
3
4


1
2
3
4
Afficher le parachute
non
parach. à la surface de l'eau
oui
partie = partie + 1
parach. sur le bateau ?
oui
Gagné = Gagné + 1
Afficher "Plouf"
Afficher les résultats

```
  10 REM RAVITAILLEMENT EN MER
  20 REM --------------------
  30 LET P=0
  40 LET G=0
  90 DIM B$(2,5)
 100 LET B$(1)="  ▄█▄"
 110 LET B$(2)="/▀██▀"
 190 DIM A$(2,5)
 200 LET A$(1)="▄    "
 210 LET A$(2)="█▄▄▄▄"
 290 DIM P$(2,2)
 300 LET P$(1)="()"
 310 LET P$(2)="▀▀"
 320 REM --------------------
 330 REM JEU
 440 LET YP=0
 450 LET XP=0
 460 LET FLAG=0
 470 LET X=1
 480 GOSUB 2000
 490 PRINT AT 0,28;"  "
 500 PRINT AT 1,28;"     "
 505 IF X=28 THEN GOTO 1200
 510 PRINT AT 0,X;A$(1)
 520 PRINT AT 1,X;A$(2)
 530 LET X=X+1
 550 IF X=28 THEN GOTO 1200
 560 IF INKEY$<>"" THEN LET FLAG
=1
 570 IF FLAG=0 THEN GOTO 510
 580 IF XP=0 THEN LET XP=X
 590 PRINT AT YP,XP;"  "
 595 PRINT AT YP+1,XP;"   "
 597 IF XP>26 THEN GOTO 1700
 600 LET YP=YP+1
 605 LET XP=XP+V
 610 PRINT AT YP,XP;P$(1)
 620 PRINT AT YP+1,XP;P$(2)
 630 IF YP=20 THEN GOTO 1400
1000 GOTO 505
1200 IF FLAG=0 THEN GOTO 440
1210 GOTO 580
1380 REM ---------------------
1390 REM GAGNE OU PERDU ..?
1400 LET P=P+1
1402 IF XP>XB AND XP<XB+4 THEN G
OTO 1500
1403 REM ---------------------
1404 REM PERDU...
1405 LET XP=XP+1
1410 FOR I=1 TO 4
1420 PLOT (XP*2)-I,((20-YP)*2)-I
1430 PLOT (XP*2)+I,((20-YP)*2)-I
1440 PLOT (XP*2)-I*2,((20-YP)*2)
-I
1450 PLOT (XP*2)+I*2,((20-YP)*2)
-I
1460 NEXT I
1470 GOTO 1600
1480 REM ---------------------
1490 REM GAGNE...
1500 LET G=G+1
1510 GOTO 1600
1580 REM ---------------------
1590 REM RESULTATS
1600 PRINT AT 10,10;"PARTIE:";P
1610 PRINT AT 12,10;"GAGNE :";G
1620 FOR I=1 TO 100
1630 NEXT I
1640 CLS
1650 GOTO 440
1680 REM --------------------
1690 REM HORS DE L"ECRAN
1700 LET P=P+1
1710 GOTO 1600
1980 REM --------------------
1990 REM AFFICHE LE BATEAU
2000 FOR I=0 TO 29
2010 PRINT AT 20,I;"_"
2020 PRINT AT 21,I;"-"
2030 NEXT I
2040 LET XB=RND*24
2050 PRINT AT 20,XB;B$(1)
2060 PRINT AT 21,XB;B$(2)
2070 LET V=INT (RND*2)/2
2075 IF V=0 THEN GOTO 2090
2080 PRINT AT 10,0;"VENT"
2090 FOR I=1 TO 50
2100 NEXT I
2110 PRINT AT 10,0;"    "
2120 RETURN
```

LE PROGRAMME

Lignes 30 à 310

On assigne respectivement, aux variables B$, A$ et P$, les dessins des bateaux, avions et parachutes.

Lignes 440 à 1210

On commence par initialiser les coordonnées de l'avion et du parachute, puis on affiche le bateau en utilisant le sous-programme des lignes 2050 à 2120.

Ce programme principal (lignes 440 à 1210) boucle sur lui-même pour afficher, sans interruption, l'avion et le parachute, tout en les déplaçant pour donner cet effet de mouvement.

Tant qu'une touche n'est pas appuyée (ligne 560), le parachute n'apparaît pas. Dans le cas contraire, la variable FLAG=1 et le parachute commence sa descente jusqu'à la surface de l'eau.

Quand le parachute est au niveau de l'eau (YP=20 en ligne 630) on saute au programme en ligne 1400 pour vérifier que le parachute est oui ou non sur le bateau.

Lignes 1400 à 1402

On incrémente le compteur de parties jouées P. Si le parachute est tombé sur le bateau, on va à la ligne 1500 (programme "GAGNE"), sinon, on continue (programme "PERDU").

Lignes 1405 à 1470

Le parachute n'étant pas arrivé sur le bateau, on dessine le jaillissement de l'eau provoqué par la chute.

Lignes 1500 à 1510

Le parachute étant sur le bateau, on incrémente le nombre de parties gagnées.

Lignes 1600 à 1650

On affiche, au milieu de l'écran, le nombre de parties gagnées avant de retourner à un nouvel essai.

Lignes 1700 à 1710

Si le parachute est sorti de l'écran, on incrémente simplement le nombre de parties jouées.

DATE	NOM DU PROGRAMME : RAVITAILLEMENT EN MER	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A$( )	TABLEAU		X				Dessin de l'avion
B$( )	TABLEAU		X				Dessin du bateau
FLAG	AIGUILLAGE		X				Flag = 1 si on lache le
							parachute
G	COMPTEUR		X				Nombre de parties gagnees
I	INDICE		X				Boucle FOR/NEXT
P	COMPTEUR		X				Nombre de parties
P$( )	TABLEAU		X				Dessin du parachute
V	VARIABLE		X				Force du vent
X	VARIABLE		X				coordonnées de l'avion
XB	VARIABLE		X				coordonnées du bateau
XP	VARIABLE		X				} coordonnées
YP	VARIABLE		X				} du parachute

6.1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Biorythme

Ce programme permet la représentation graphique du biorythme de chacun pour un mois donné.

Les courbes obtenues dessinent les trois principaux cycles qui influencent la vie humaine, avec leurs périodes positives et négatives : le cycle physique de 23 jours, le cycle émotionnel de 28 jours et le cycle intellectuel de 31 jours. Ces données, tout-à-fait sérieuses, ont fait l'objet de longues études statistiques.

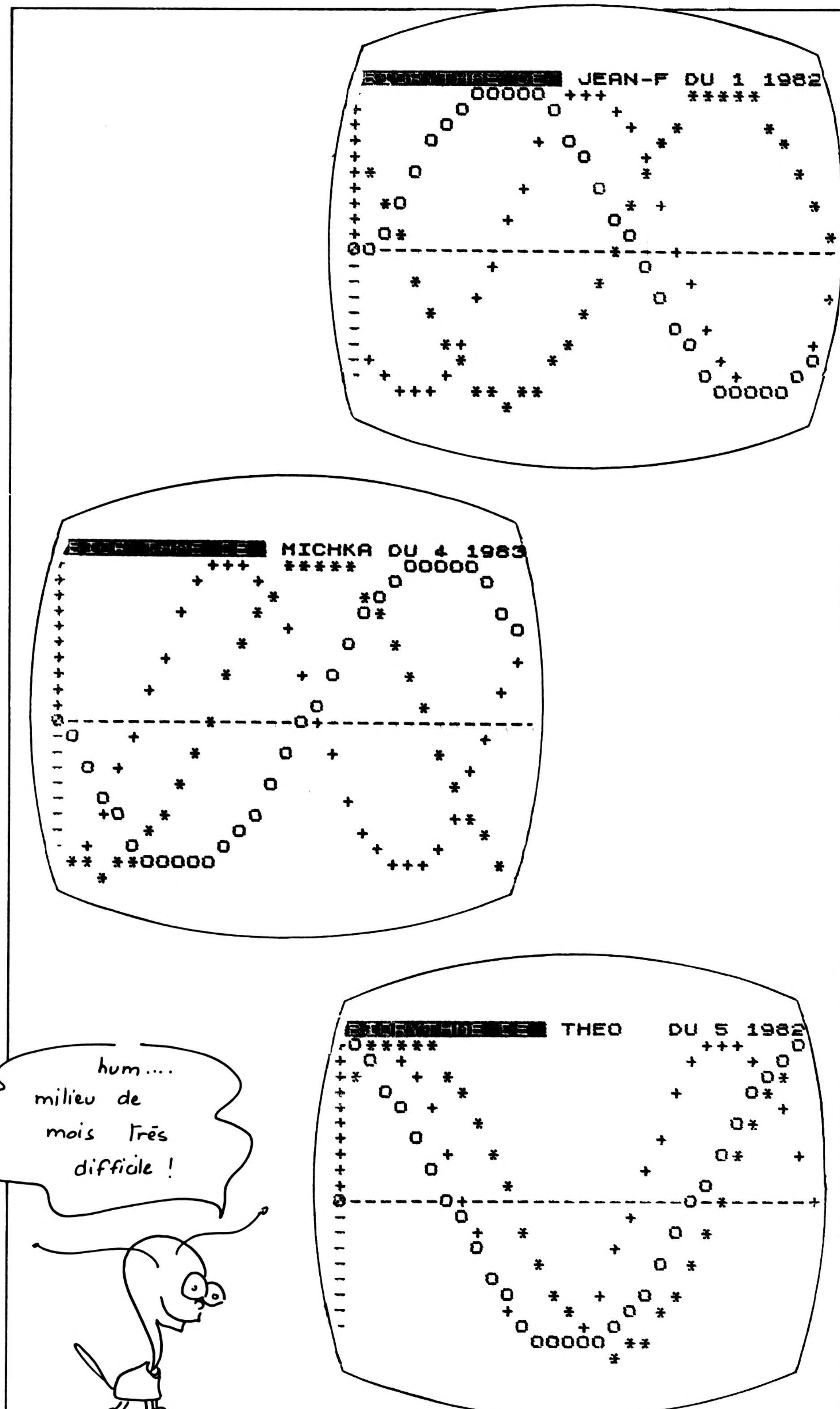
Pendant les périodes positives, notre énergie se libère et nous sommes actifs. Pendant les périodes négatives, notre organisme récupère ses forces. Aux points zéro des courbes, nous vivons des jours critiques.

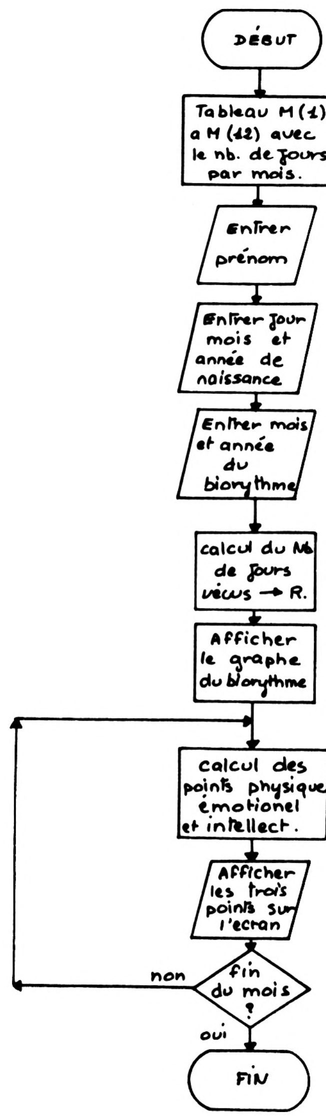
Ce ne sont pas forcément des jours dangereux, mais ce sont sûrement des jours où il faudra redoubler d'attention. Une grande prudence est nécessaire pendant les jours critiques doubles (deux courbes aux points zéro), et il est vivement conseillé de ne pas sortir de chez soi les jours critiques triples...

Pour calculer votre biorythme, le ZX81 demandera successivement le prénom (sur 6 lettres maximum), le jour de naissance, puis le mois et l'année, le mois du biorythme souhaité, puis l'année. Toutes ces données sont à entrer en tapant sur **NEWLINE**.

Le biorythme se dessine alors sur l'écran.

Les possesseurs d'imprimante pourront en garder une recopie d'écran en utilisant l'instruction **COPY**.


JEAN-F DU 1 1982
MICHKA DU 4 1983
THEO DU 5 1982
hum.... milieu de mois trés difficile !


DÉBUT
Tableau M(1) a M(12) avec le nb. de jours par mois.
Entrer prénom
Entrer jour mois et année de naissance
Entrer mois et année du biorythme
calcul du Nb de jours vécus → R.
Afficher le graphe du biorythme
calcul des points physique émotionel et intellect.
Afficher les trois points sur l'ecran
fin du mois ?
non
oui
FIN

```
  10 REM BIORYTHME
  20 REM -----------------------
  90 DIM Z$(6)
 100 DIM M(12)
 110 LET M(1)=31
 120 LET M(2)=28
 130 LET M(3)=31
 140 LET M(4)=30
 150 LET M(5)=31
 160 LET M(6)=30
 170 LET M(7)=31
 180 LET M(8)=31
 190 LET M(9)=30
 200 LET M(10)=31
 210 LET M(11)=30
 220 LET M(12)=31
 230 LET R=0
 235 REM -----------------------
 236 REM SAISIE BIORYTHME
 240 PRINT AT 19,0;"      "
 250 PRINT AT 20,0;"      "
 260 PRINT AT 21,0;"      "
 310 PRINT AT 18,0;"PRENOM ?"
 320 INPUT Z$
 330 PRINT AT 18,0;"JOUR DE NAIS
SANCE ?"
 340 INPUT T
 350 PRINT AT 18,0;"MOIS DE NAIS
SANCE ?"
 360 INPUT S
 370 PRINT AT 18,0;"ANNEE DE NAI
SSANCE ?"
 380 INPUT U
 390 PRINT AT 18,0;"MOIS DU BIOR
YTHME ? "
 400 INPUT V
 410 PRINT AT 18,0;"ANNEE DU BIO
RYTHME ?"
 420 INPUT W
 430 CLS
 440 REM -----------------------
 450 REM CALCUL DU NOMBRE DE
JOURS VECUS
 500 IF S>2 THEN GOTO 550
 510 IF S=2 AND T=29 THEN GOTO 5
50
 520 IF INT ((U-1900)/4)<>(U-190
0)/4 THEN GOTO 550
 530 LET R=1
 550 LET R=R+M(S)-T
 560 IF S=12 THEN GOTO 600
 570 FOR X=S+1 TO 12
 580 LET R=R+M(X)
 590 NEXT X
 600 IF W-U<2 THEN GOTO 650
 610 FOR X=U-1899 TO W-1901
 620 IF INT (X/4)=X/4 THEN LET R
=R+1
 630 LET R=R+365
 640 NEXT X
 650 IF V=1 THEN GOTO 700
 660 FOR X=1 TO V-1
 670 LET R=R+M(X)
 680 NEXT X
 700 IF V>2 AND INT ((W-1900)/4)
=(W-1900)/4 THEN LET R=R+1
 710 REM -----------------------
 720 REM EDITION DU BIORYTHME
 800 PRINT " BIORYTHME DE  ";Z$;
" DU ";V;" ";W+1900
 810 PRINT
 820 FOR I=1 TO 10
 830 PRINT AT I,0;"+"
 840 PRINT AT I+11,0;"-"
 850 NEXT I
 860 PRINT AT 11,0;"0"
 870 FOR I=1 TO 31
 880 PRINT AT 11,I;"-"
 890 NEXT I
1000 LET A=M(V)
1010 LET B=0
1020 LET B=B+1
1030 LET C=INT (10*SIN ((B+R)*(2
*PI/23))+11)
1040 LET D=INT (10*SIN ((B+R)*(2
*PI/28))+11)
1050 LET E=INT (10*SIN ((B+R)*(2
*PI/33))+11)
1100 PRINT AT C,B;"+"
1110 PRINT AT D,B;"*"
1120 PRINT AT E,B;"O"
1150 IF B<A THEN GOTO 1020
```

LE PROGRAMME

Lignes 90 à 230

Après dimensionnement du tableau M(), on y place le nombre de jours de chaque mois.

EXEMPLE : janvier (1er mois de 31 jours) : M(1) = 31.

Lignes 236 à 430

On saisit, dans ces lignes, tous les renseignements nécessaires à l'élaboration du biorythme. Pour préciser l'endroit où sont tapés ces renseignements, les lignes 240 à 260 dessinent une flèche désignant le bas de l'écran.

Lignes 450 à 700

On calcule, dans ces lignes, le nombre de jours vécus par l'intéressé, en tenant compte des années bissextiles évidemment.

Lignes 800 à 1150

Avec les renseignements collectés ci-dessus, on peut éditer le biorythme. Après les axes, on dessine, point par point, les trois courbes en calculant leurs ordonnées par une fonction sinus.

Leurs abcisses correspondent aux numéros des jours concernés.

DATE	NOM DU PROGRAMME : BIORYTHME	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A	VARIABLE		X				Nombre de Jours du mois à
							étudier
B	COMPTEUR		X				Nombre de Jours déjà affichés
C	VARIABLE		X				Coordonnées du point " Physique "
							sur la courbe
D	VARIABLE		X				Coordonnées du point " Emotionel "
							sur la courbe
E	VARIABLE		X				Coordonnées du point " intellect "
							sur la courbe
I	INDICE		X				Boucle FOR / NEXT de l'édition
							du biorythme
M ( )	TABLEAU		X				Contient le nombre de Jours par mois
R	VARIABLE		X				Nombre de Jours vécus
S	VARIABLE	X					Mois de Naissance
T	VARIABLE	X					Jour de naissance
U	VARIABLE	X					année de naissance
V	VARIABLE	X					Mois du biorythme
W	VARIABLE	X					année du biorythme
X	INDICE		X				Boucle FOR / NEXT pour le
							calcul du nombre de Jours vécus.
Z $ ( )	TABLEAU	X					prénom.

6.1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Macro-lettres

Pour créer la banderole en gros caractères du texte que vous voulez, utilisez le programme présenté ici.

Il suffira, après avoir tapé **RUN**, de donner votre message en bas de l'écran.

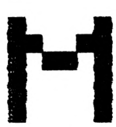
Votre ZX81, possédant sa matrice de caractères en mémoire morte (ROM8K), il est possible d'y accéder facilement.

Une matrice de caractères est ni plus ni moins que l'image informatique des caractères dessinés sur l'écran. Chaque point est défini dans la mémoire par des 1 et des 0, le 1 correspondant à un point, le 0 à l'absence de point.



```
011110
100001
100000
100000
100001
011110
```

```
 1111
1    1
1
1
1    1
 1111
```



```
100001
110011
101101
100001
100001
100001
```

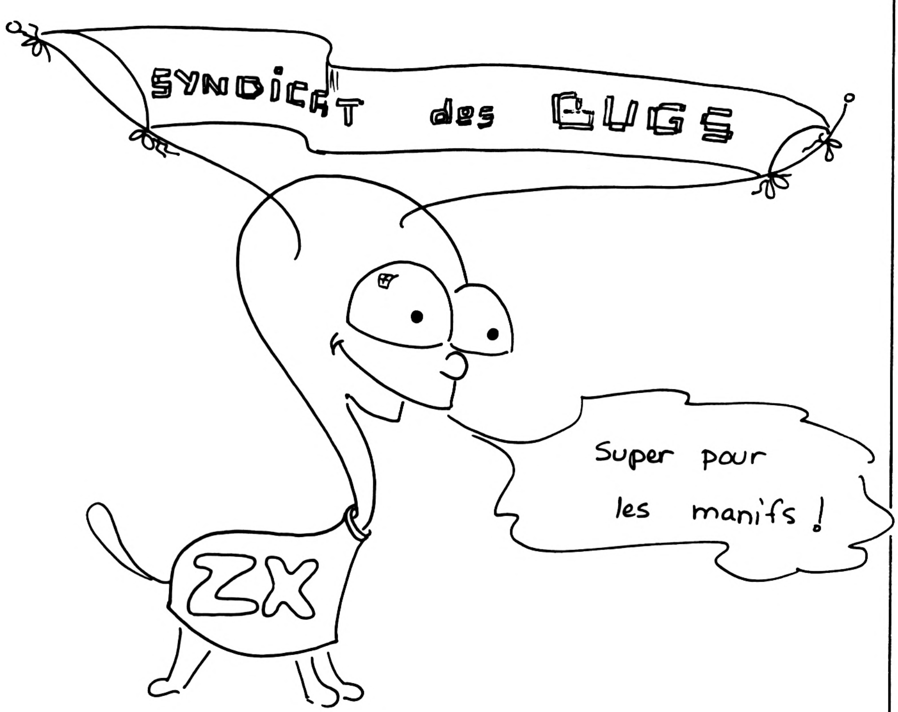
```
1    1
11  11
1 11 1
1    1
1    1
1    1
```

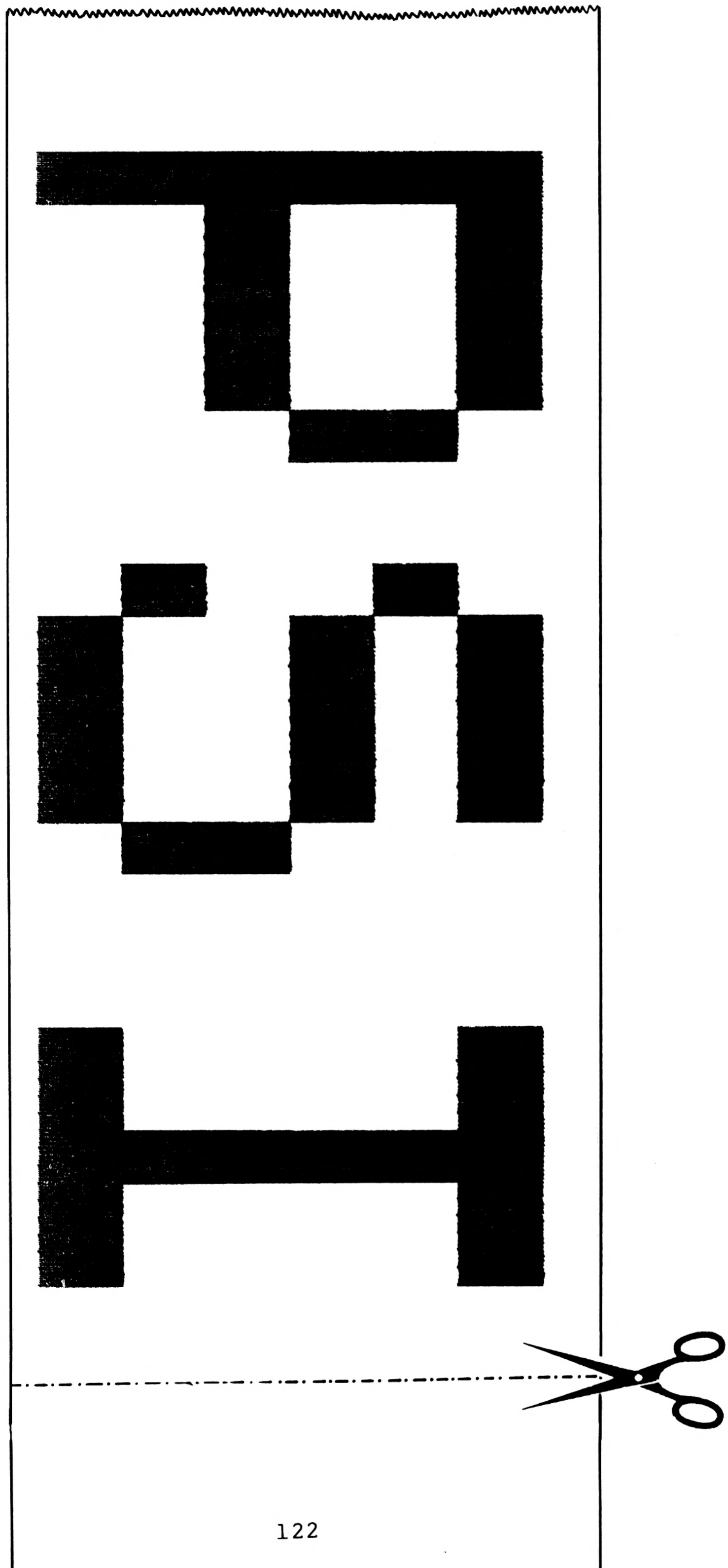


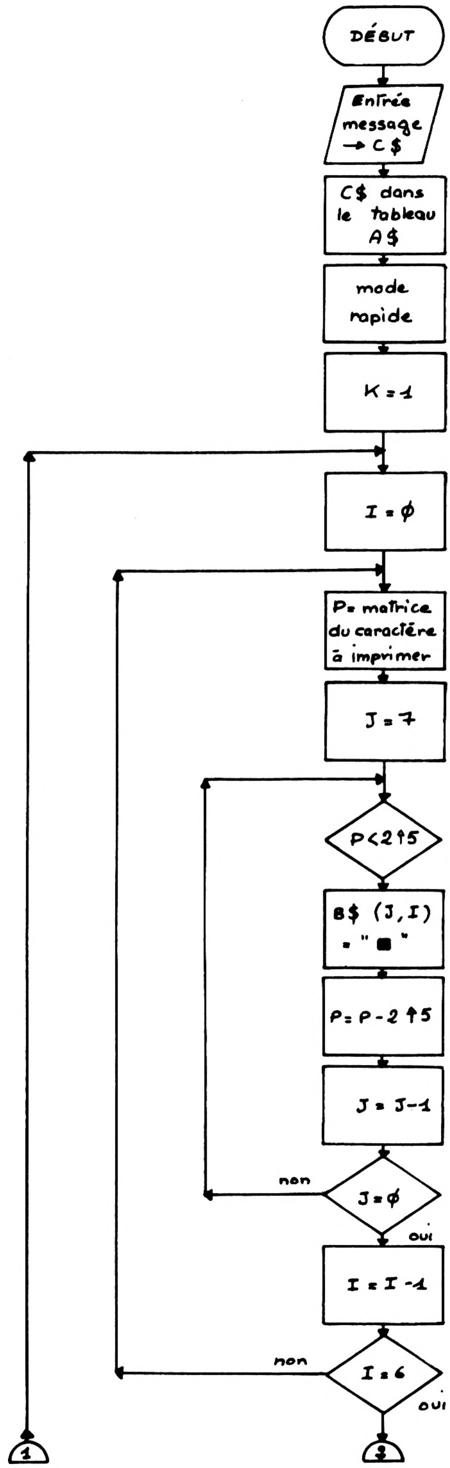
```
11111
1    1
1    1
11111
1
1
```

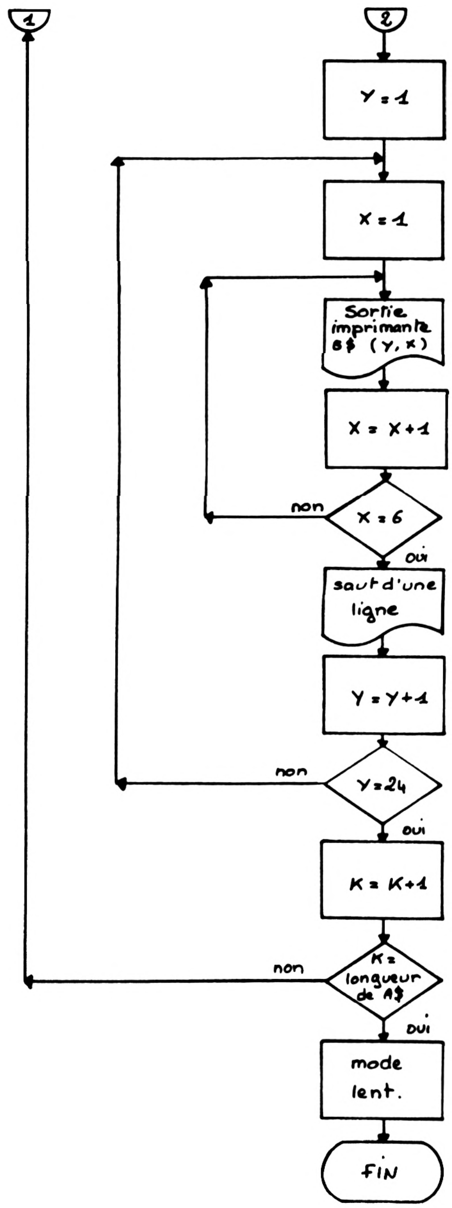
```
111110
100001
100001
111110
100000
100000
```

```
 10 REM MACRO LETTRES
 20 REM --------------------
 30 INPUT A$
 40 FAST
 50 REM --------------------
 55 REM CALCUL
 60 FOR K=1 TO LEN A$
 70 DIM B$(24,6,5)
 80 FOR I=0 TO 6
 90 LET P=PEEK (7680+(8*CODE A$
(K))+I)
100 FOR J=7 TO 0 STEP -1
120 IF P<2**J THEN GOTO 150
130 LET B$((8-J)*3+0,(7-I))="██
███"
135 LET B$((8-J)*3+1,(7-I))="██
███"
137 LET B$((8-J)*3+2,(7-I))="██
███"
140 LET P=P-2**J
150 NEXT J
160 NEXT I
170 REM --------------------
180 REM IMPRESSION
190 FOR Y=1 TO 24
200 FOR X=1 TO 6
210 LPRINT B$(Y,X);
220 NEXT X
230 LPRINT
240 NEXT Y
250 NEXT K
260 SLOW
```






DÉBUT
Entrée message → C$
C$ dans le tableau A$
mode rapide
K = 1
I = ∅
P = matrice du caractère à imprimer
J = 7
P<2↑5
B$ (J,I) = "■"
P = P-2↑5
J = J-1
non
J = ∅
oui
I = I-1
non
I = 6
oui
1
2


1
2
Y = 1
X = 1
Sortie imprimante B$ (Y, X)
X = X+1
non
X = 6
oui
saut d'une ligne
Y = Y+1
non
Y = 24
oui
K = K+1
non
K = longueur de A$
oui
mode lent.
FIN

LE PROGRAMME

Ligne 30

On saisit, au clavier, le message à imprimer sur la bande papier. Ce message peut avoir n'importe quelle longueur. Pour gagner du temps dans les calculs qui vont suivre, on passe en mode rapide (**FAST**).

Lignes 60 à 160

Pour générer les caractères, on utilise trois boucles imbriquées.

La première boucle prélève un à un les caractères du message. La deuxième boucle prélève les sept lignes de la matrice de caractères. La troisième boucle décompose les huit points de chaque ligne.

La matrice de caractères du ZX81, étant placée dans la mémoire morte (**ROM**), il est inutile de récréer l'image de ces caractères. Par des **PEEK** dans cette mémoire (la matrice commence à l'adresse 7680), on extrait chaque point à imprimer, en sachant que zéro correspond à un point blanc, et un à un point noir. Chaque point est placé dans le tableau B$ et agrandi 5 fois.

On utilise un tableau pour obtenir des caractères plus grands que l'écran, ce qui est possible sur le papier, celui-ci étant infini sur sa longueur !

On retourne bien sûr ces caractères de 270 degrés pour qu'ils soient lisibles, c'est-à-dire non inversés (comme dans une glace) et écrits de gauche à droite.

Lignes 190 à 260

Le tableau B$ contenant un caractère, on l'imprime sur la bande papier par l'instruction **LPRINT**.

Dès que tous les caractères sont imprimés, on revient en mode lent (**SLOW**).

DATE	NOM DU PROGRAMME : MACRO - LETTRES	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	Programme	Cassette	Disquette	LONG.	OBSERVATIONS
A$( )	TABLEAU		X				Message à écrire
B$( )	TABLEAU		X				Lettre à écrire
C$	VARIABLE	X					Message à écrire
I	INDICE		X				Boucle FOR/NEXT
J	INDICE		X				Boucle FOR/NEXT
K	INDICE		X				Boucle FOR/NEXT
P	VARIABLE		X				Matrice du caractère à imprimer
X	INDICE		X				Boucle FOR/NEXT
Y	INDICE		X				Boucle FOR/NEXT

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Fichier téléphone

Pourquoi ne pas simplifier votre annuaire et vous réserver quelques numéros de téléphone ?

Alors, voici un petit fichier qui vous permettra de garder les numéros d'une vingtaine de personnes.

PREMIERE UTILISATION :

Après avoir tapé **RUN**, vingt lignes s'affichent.

L'ordinateur vous demande le numéro de la ligne à remplir, puis le nom, puis le numéro de téléphone. Pour modifier une ligne, il faudra réécrire sur cette même ligne. Pour supprimer une ligne entièrement ou non, il faudra remplacer ce qui est écrit par des espaces.

Si vous voulez sauvegarder votre fichier sur cassette, vous donnez n'importe quel numéro, puis vous marquez "FIN" comme nom.

Le message suivant apparaît :

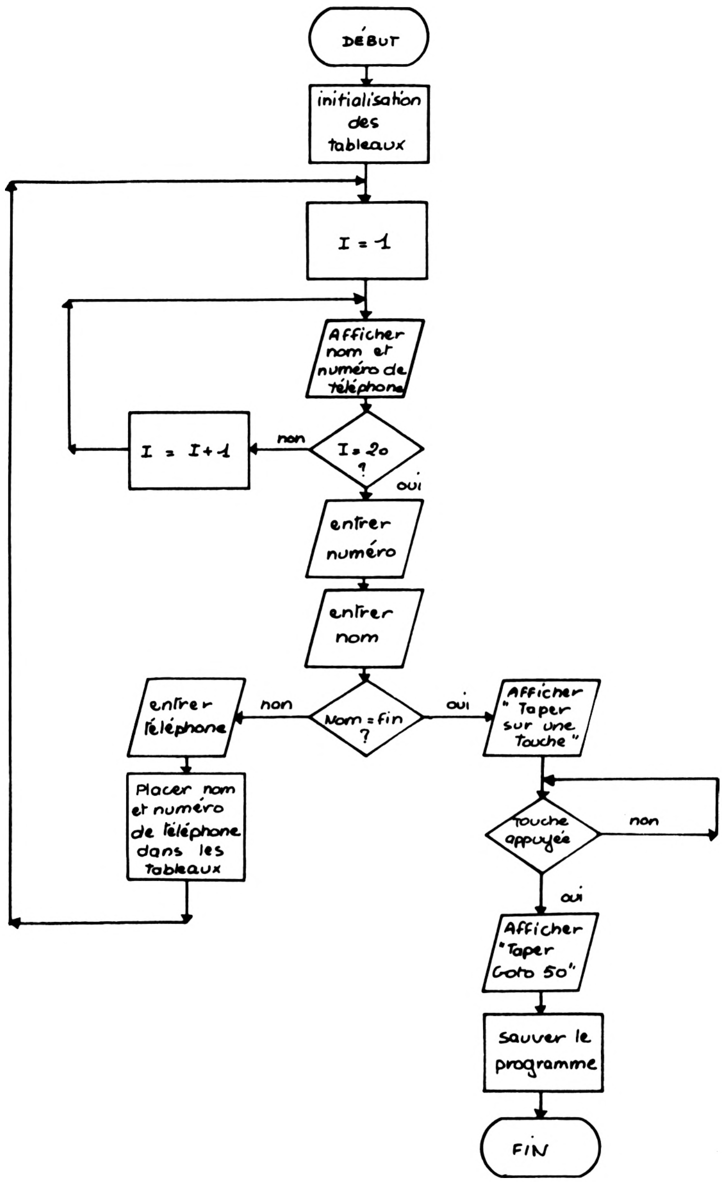
"MAGNETO O.K ? TAPEZ SUR UNE TOUCHE"

Vous devez mettre votre magnétophone en position d'engistrement. Dès qu'il est prêt, vous tapez sur n'importe quelle touche et la sauvegarde complète du programme et du contenu des tableaux s'effectue.

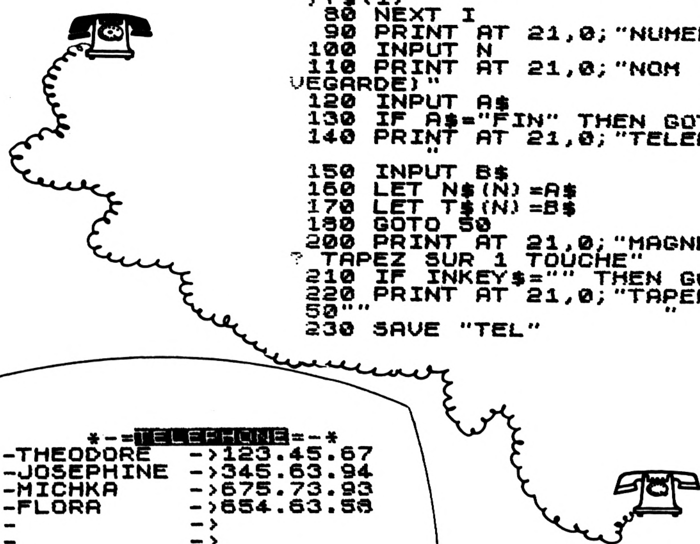
DEUXIEME ET AUTRES UTILISATIONS :

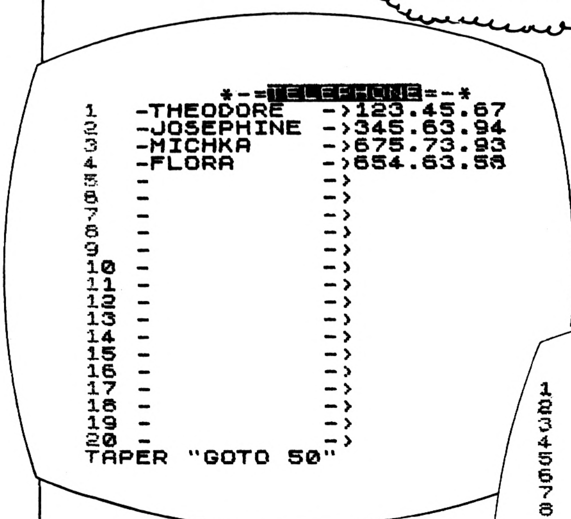
Quand vous chargez à nouveau votre "programme et tableaux", le message "**GOTO 50**" apparaît. N'oubliez pas de le taper, sinon vous perdriez toutes les informations de votre fichier. En effet, la commande **RUN** ferait repartir le programme à zéro en vidant les tableaux.

Bien sûr, vous pouvez avoir un listing sur papier de vos numéros en faisant **BREAK** puis **COPY**. Ne jamais utiliser **RUN** pour repartir, mais "**GOTO 50**".

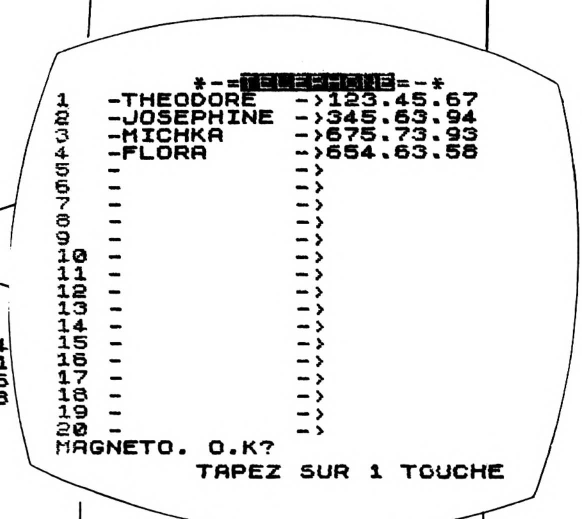

DÉBUT
initialisation des tableaux
I = 1
Afficher nom et numéro de téléphone
I = 20 ?
non
oui
I = I+1
entrer numéro
entrer nom
Nom = fin ?
non
oui
entrer téléphone
Placer nom et numéro de téléphone dans les Tableaux
Afficher " Taper sur une Touche "
Touche appuyée
non
oui
Afficher "Taper Goto 50"
Sauver le programme
FIN

```
 10 REM FICHIER TELEPHONIQUE
 20 REM ------------------
 30 DIM T$(20,9)
 40 DIM N$(20,10)
 50 CLS
 55 PRINT TAB 8;"*-=TELEPHONE=-*"
 60 FOR I=1 TO 20
 70 PRINT I;TAB 3;"-";N$(I);"->";T$(I)
 80 NEXT I
 90 PRINT AT 21,0;"NUMERO"
100 INPUT N
110 PRINT AT 21,0;"NOM (FIN=SAUVEGARDE)"
120 INPUT A$
130 IF A$="FIN" THEN GOTO 200
140 PRINT AT 21,0;"TELEPHONE   "
150 INPUT B$
160 LET N$(N)=A$
170 LET T$(N)=B$
180 GOTO 50
200 PRINT AT 21,0;"MAGNETO. O.K? TAPEZ SUR 1 TOUCHE"
210 IF INKEY$="" THEN GOTO 210
220 PRINT AT 21,0;"TAPER ""GOTO 50""      "
230 SAVE "TEL"
```

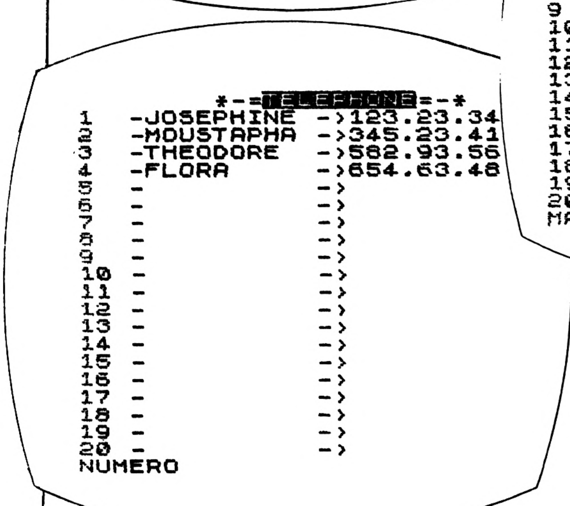




```
        *-=TELEPHONE=-*
1  -THEODORE   ->123.45.67
2  -JOSEPHINE  ->345.63.94
3  -MICHKA     ->675.73.93
4  -FLORA      ->654.63.58
5  -           ->
6  -           ->
7  -           ->
8  -           ->
9  -           ->
10 -           ->
11 -           ->
12 -           ->
13 -           ->
14 -           ->
15 -           ->
16 -           ->
17 -           ->
18 -           ->
19 -           ->
20 -           ->
TAPER "GOTO 50"
```



```
        *-=TELEPHONE=-*
1  -THEODORE   ->123.45.67
2  -JOSEPHINE  ->345.63.94
3  -MICHKA     ->675.73.93
4  -FLORA      ->654.63.58
5  -           ->
6  -           ->
7  -           ->
8  -           ->
9  -           ->
10 -           ->
11 -           ->
12 -           ->
13 -           ->
14 -           ->
15 -           ->
16 -           ->
17 -           ->
18 -           ->
19 -           ->
20 -           ->
MAGNETO. O.K?
        TAPEZ SUR 1 TOUCHE
```



```
        *-=TELEPHONE=-*
1  -JOSEPHINE  ->123.23.34
2  -MOUSTAPHA  ->345.23.41
3  -THEODORE   ->582.93.56
4  -FLORA      ->654.63.48
5  -           ->
6  -           ->
7  -           ->
8  -           ->
9  -           ->
10 -           ->
11 -           ->
12 -           ->
13 -           ->
14 -           ->
15 -           ->
16 -           ->
17 -           ->
18 -           ->
19 -           ->
20 -           ->
NUMERO
```

LE PROGRAMME

Lignes 30 et 40

Ces deux lignes dimensionnent les tableaux des noms (N$) et des téléphones (T$). On limite le nombre d'éléments à 20, ce qui correspond au nombre de lignes sur l'écran (en laissant toutefois un peu de place pour la saisie du fichier !).

Le dimensionnement de tableaux (instruction DIM) efface le contenu des tableaux portant les mêmes noms.

Après chargement d'un programme contenant déjà des éléments des tableaux, il faudra sauter cette étape et passer directement aux lignes suivantes.

La commande de lancement ne sera plus **RUN**, mais "**GOTO 50**".

Lignes 50 à 180

Après effacement de l'écran, on affiche, un à un, les éléments du fichier. Puis l'on saisit le numéro de l'"enregistrement" que l'on veut modifier ; puis le nom et, enfin, son numéro de téléphone.

Après rangement de ces éléments dans les deux tableaux, on retourne en ligne 50 pour afficher de nouveau le fichier.

Lignes 200 à 230

Si l'on tape le mot "FIN" à la place d'un nom, le test de la ligne 130 renvoie à la partie sauvegarde du programme.

Pour vous permettre de placer votre magnétophone en mode enregistrement, on affiche un message d'attente et l'on teste si une touche est enfoncée par la fonction **INKEY$**.

Une sauvegarde de programme enregistre aussi le contenu de l'écran et le restitue au moment du chargement.

Pour que vous n'oubliez pas de taper "**GOTO 50**" au lieu de **RUN**, lors d'un chargement ultérieur, on envoie ce message à l'écran.

DATE	NOM DU PROGRAMME : FICHIER TÉLÉPHONIQUE	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A$	VARIABLE	X					Nom à saisir
B$	VARIABLE	X					Numéro de téléphone
							à saisir
I	INDICE		X				Boucle FOR/NEXT pour
							l'édition du tableau
N	VARIABLE	X					Numéro dans le fichier
N$( )	TABLEAU		X				Tableau des noms
T$( )	TABLEAU		X				Tableau des numéros de
							téléphone.

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Gestion de fichier

Ce programme permet la gestion et la sauvegarde, sur cassette, d'un fichier d'une centaine d'articles.

La première fois que vous utiliserez ce programme, vous tapez **RUN** pour le lancer. Mais, dès qu'un article sera rentré en mémoire, et si vous interrompez l'exécution par **BREAK** ou sauvegarde et arrêt d'exécution pour la reprendre ensuite, il faudra taper **GOTO 500**. En effet, **RUN** réinitialisant à vide les tableaux définis par **DIM**, toute information précédemment rentrée serait perdue.

Au lancement du programme (la 1ère ou les autres fois), un MENU s'affiche, signalant toutes les possibilités de travail : création, consultation, suppression, listing, édition, sauvegarde. Pour l'obtenir, vous devez taper le numéro du travail demandé : 1, 2, 3, 4, 5 ou 6.

CREATION

Vous pouvez rentrer le nom, le prénom, la rue (adresse), le code postal et la ville d'un individu.

Après avoir écrit chaque renseignement, tapez sur **NEWLINE**. Après avoir rentré le nom, le ZX81 trie son fichier, rangeant le nom par ordre alphabétique, d'où, parfois, le message d'attente : "JE TRIE".

TRES IMPORTANT : Il ne faut jamais utiliser la touche espace dans les données d'un article :

Exemple :

~~DE BIZET~~ faux
~~25 AV DE LA GARE~~ faux

En effet, le ZX81 considérera la valeur de cette touche comme un BREAK et arrêtera l'exécution du programme. Il conviendra de donc d'utiliser des points "." comme séparateurs.

Exemple :

DE.BIZET
25.AV.DE.LA.GARE

Au cas où toutefois, à tout moment de l'exécution, vous tapez **BREAK**, vous pouvez repartir en tapant **GOTO 500** (et surtout pas RUN). Si vous aviez rentré le nom de l'article, celui-ci sera déjà enregistré en mémoire, mais le reste de l'article sera vide.

Comme l'option modification n'existe pas, il faudra supprimer l'article et en créer un autre ; même chose pour toute autre modification sur l'adresse.

Tous les articles se comparent sur le nom ; quand vous créez un article qui existe déjà, un message "EXISTE DEJA" apparaît.

A la fin de l'affichage complet d'une création, le ZX81 vous propose une autre création ; si vous ne la souhaitez pas, retapez sur **NEWLINE** et vous retournerez au MENU. La démarche sera la même pour la SUPPRESSION et la CONSULTATION.

CONSULTATION

Après avoir donné un nom, la fiche entière apparaît. Si votre article n'a pas été créé, un message "INEXISTANT" vous répond.

SUPPRESSION

La suppression d'un article s'effectue après en avoir rentré le nom. Si votre article n'a pas été créé, un message "INEXISTANT" apparaît, tout comme pour la consultation.

LISTING

Il s'agit du listing sur l'écran en une ou plusieurs pages du fichier (nom et prénom seulement) par ordre alphabétique.

EDITION

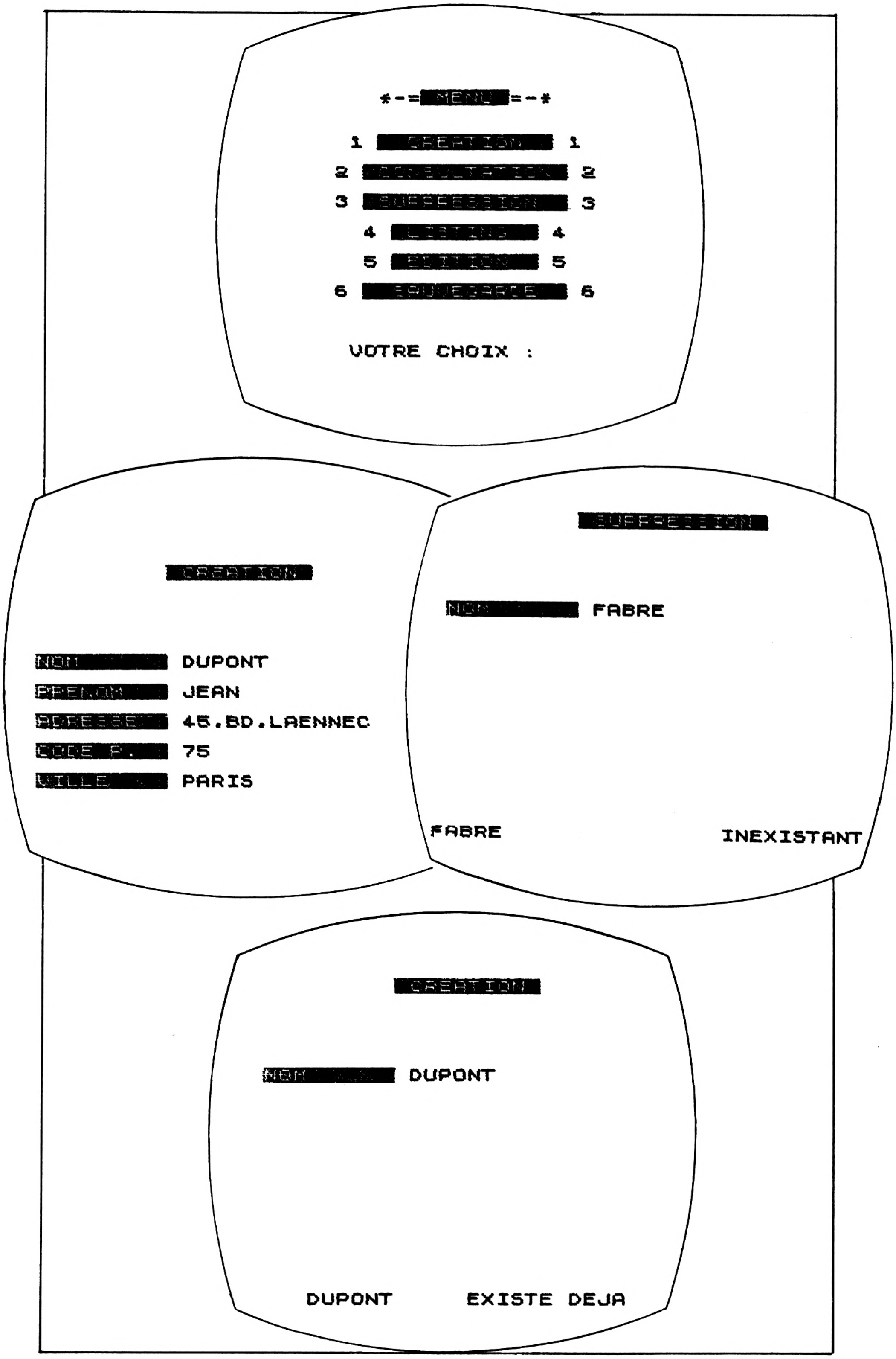
Vous obtiendrez, sur papier, tous les articles complets du fichier.

SAUVEGARDE

Cela permet la sauvegarde du fichier sur cassette (remplissage des tableaux). La marche à suivre est indiquée sur l'écran.

En cas de demande de sauvegarde involontaire, taper **BREAK**, puis **GOTO 500** pour repartir.

Pour plus d'explications sur le procédé de la sauvegarde, lisez les explications du programme "FICHIER-TELEPHONE".


-= MENU =-
1 CREATION 1
2 CONSULTATION 2
3 SUPPRESSION 3
4 LISTING 4
5 EDITION 5
6 SAUVEGARDE 6
VOTRE CHOIX :
CREATION
NOM DUPONT
PRENOM JEAN
ADRESSE 45.BD.LAENNEC
CODE P. 75
VILLE PARIS
SUPPRESSION
NOM FABRE
FABRE INEXISTANT
CREATION
NOM DUPONT
DUPONT EXISTE DEJA

```
API            AUGUSTE
BECHU          NICOLE
BONET          SOPHIE
CHRIS          CHANTAL
DATUI          PAUL
DAUDET         LEON
DIMO           LOUIS
DUPONT         JEAN
FOUGERES       LEA
GOL            MARCEL
ISTRES         CLAUDE
LALANDE        DANIELE
LION           DAVID
MENUI          ALPHONSE
NIHIL          SEXTUS
OLIVE          ARMELLE
RIVIERE        LUC
ROCHE          DENISE
SANTA          YVES

TAPEZ UNE TOUCHE POUR LA SUITE
```

```
SIMO           MARTINE
TOLSTOI        IVAN
VASCO          LOUISE
WALTER         JOAN

TAPEZ UNE TOUCHE POUR LE MENU
```

```
*-= LISTING FICHIER =-*

  API           AUGUSTE
  ROND.POINT
  44     NANTES

  BECHU         NICOLE
  LES.PINS
  53     LAVAL

  BONET         SOPHIE
  3.AV.DE.LA.GARE
  68     GRENOBLE

  CHRIS         CHANTAL
  RUE.COLLY
  37     TOURS

  DATUI         PAUL
  GARE
  62     PASO

  DAUDET        LEON
  MOULIN
  13     MARSEILLE

  DIMO          LOUIS
  PLACE.ROUGE
  35     RENNES

  DUPONT        JEAN
  45.BD.LAENNEC
  75     PARIS

  FOUGERES      LEA
  RUE.VERTE
  35     GUICHEN

  GOL           MARCEL
  RUE.KENNEDY
  59     LILLE

  ISTRES        CLAUDE
  2.RUE.JEANNE.D.ARC
  75     PARIISS

  LALANDE       DANIELE
  PLACE.BLEUE
  75     PARIS

  LION          DAVID
  RUE.MONGE
  35     RENNES

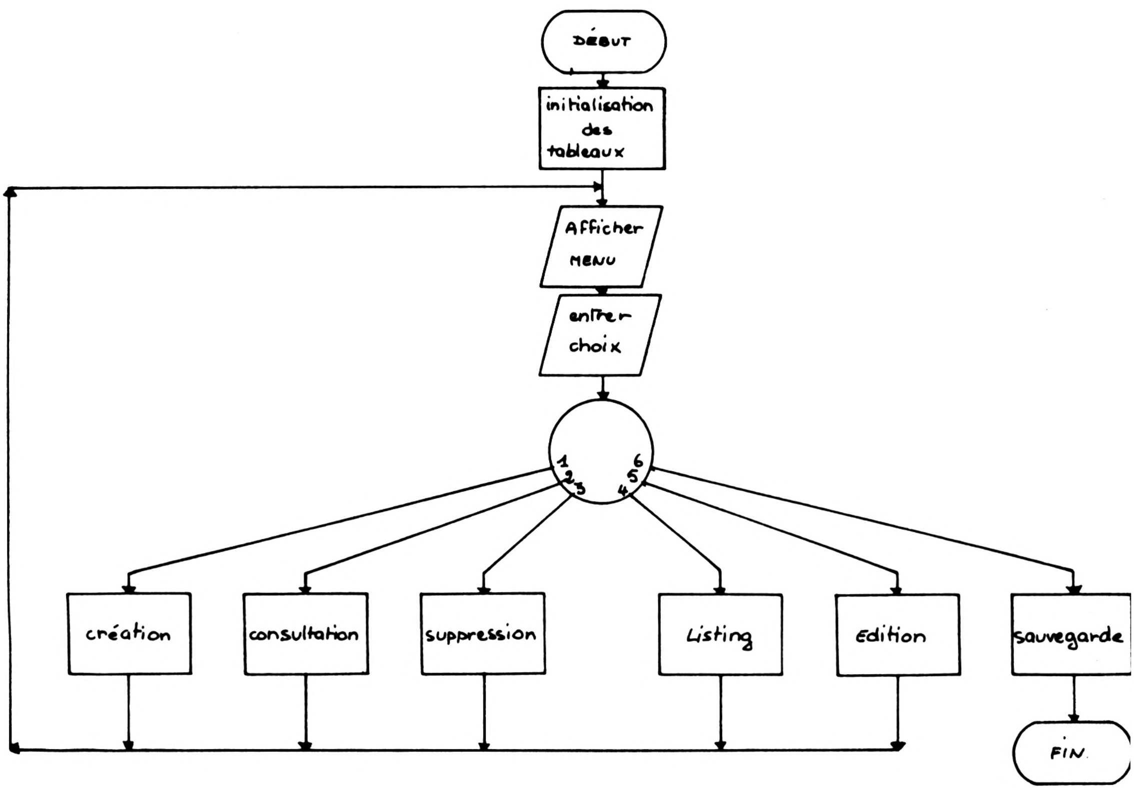
  MENUI         ALPHONSE
  HLM.3
  60     MERU

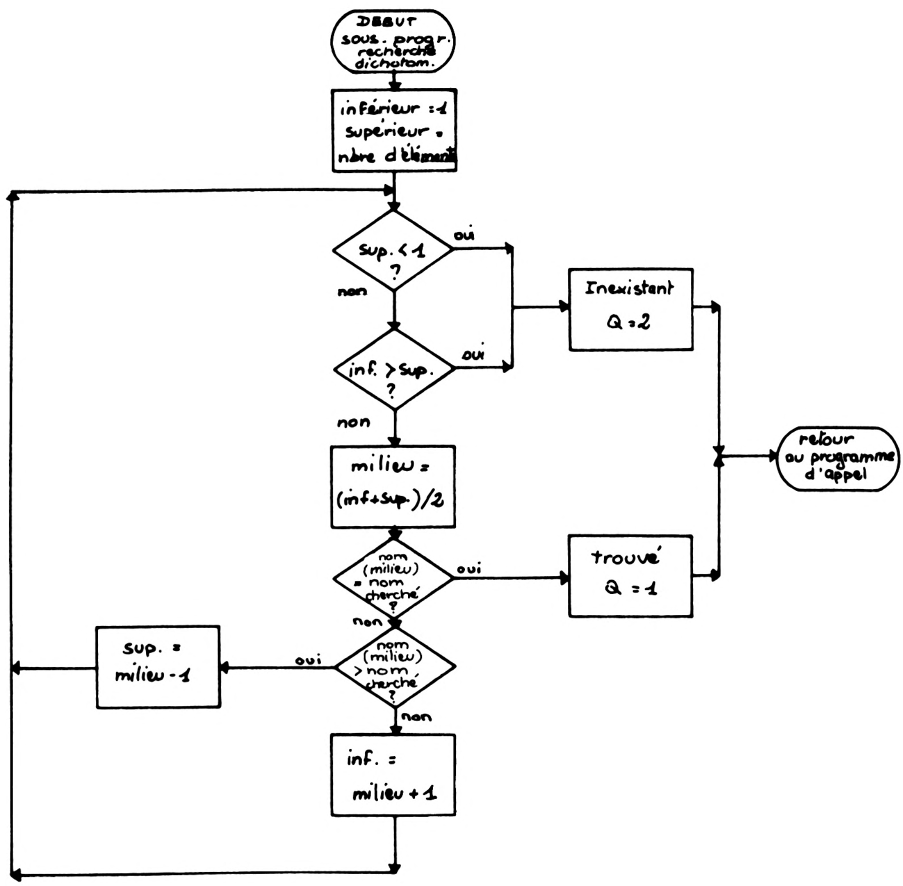
  NIHIL         SEXTUS
  LE.COLISEE
         ROME

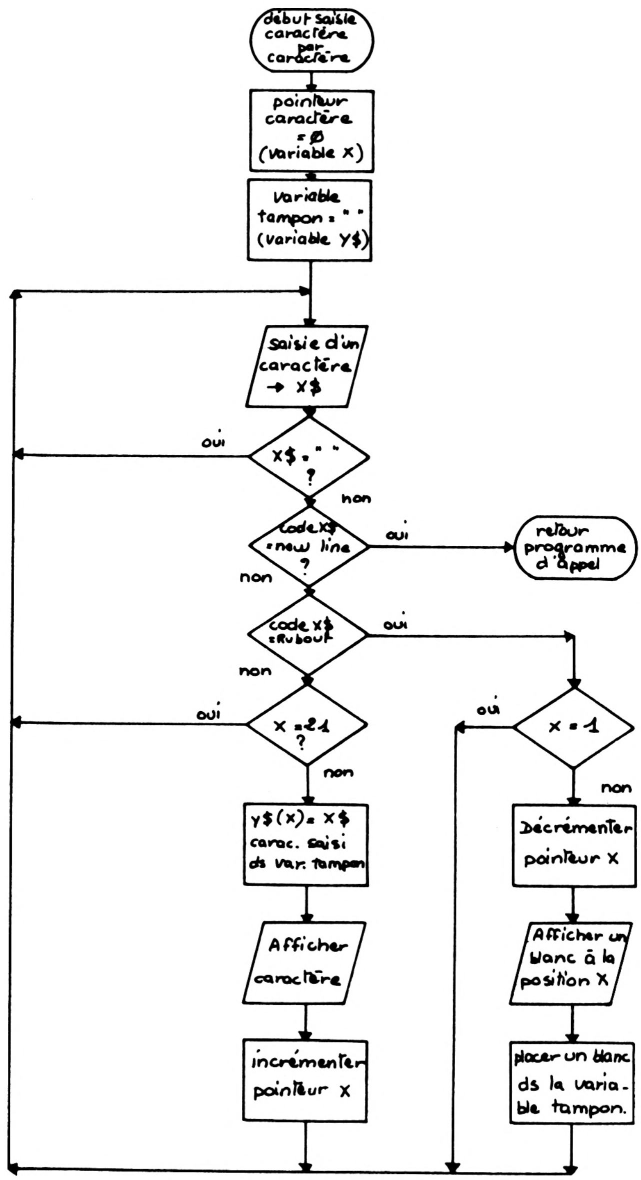
  OLIVE         ARMELLE
  LA.GARE
  45     ORLEANS

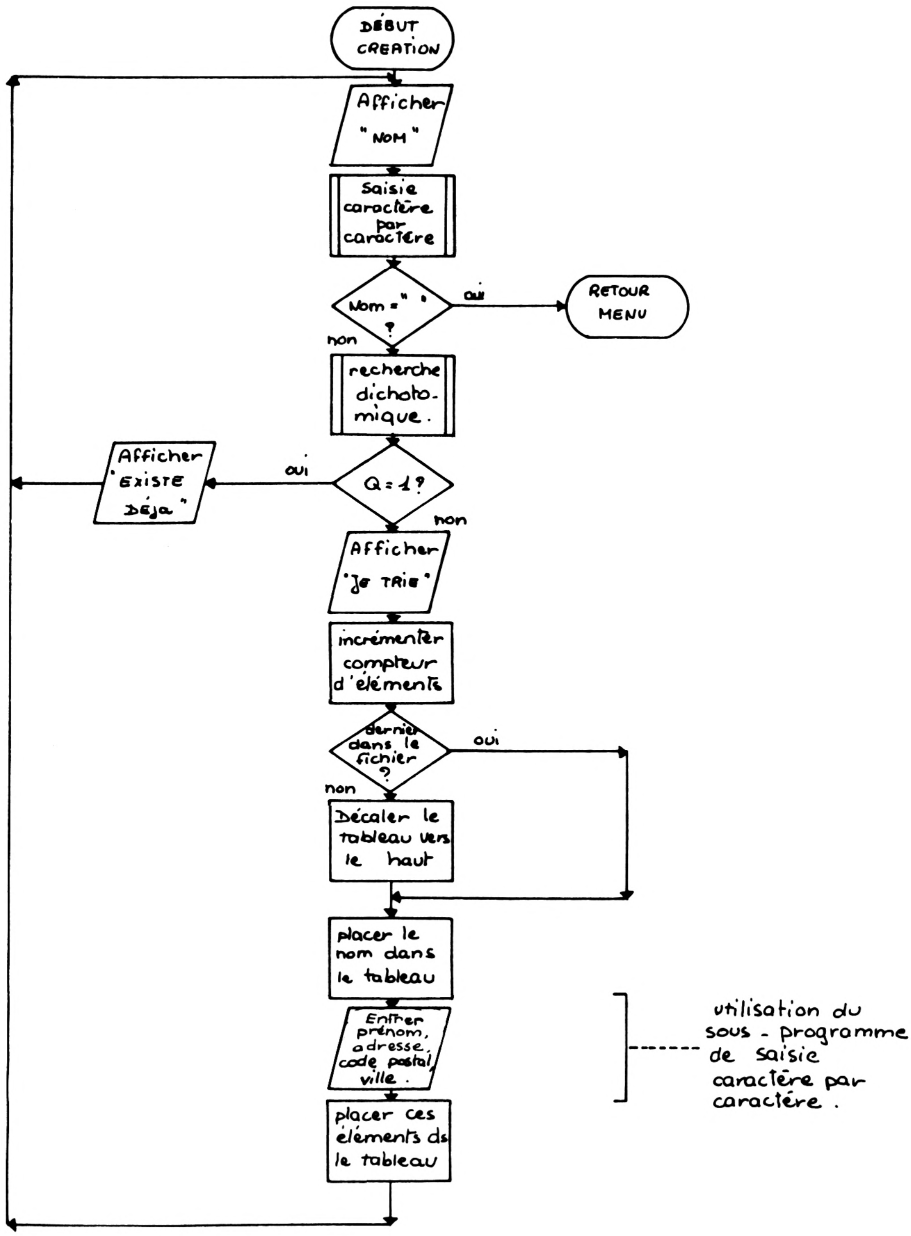
  RIVIERE       LUC
  SP.69
         RDA
```

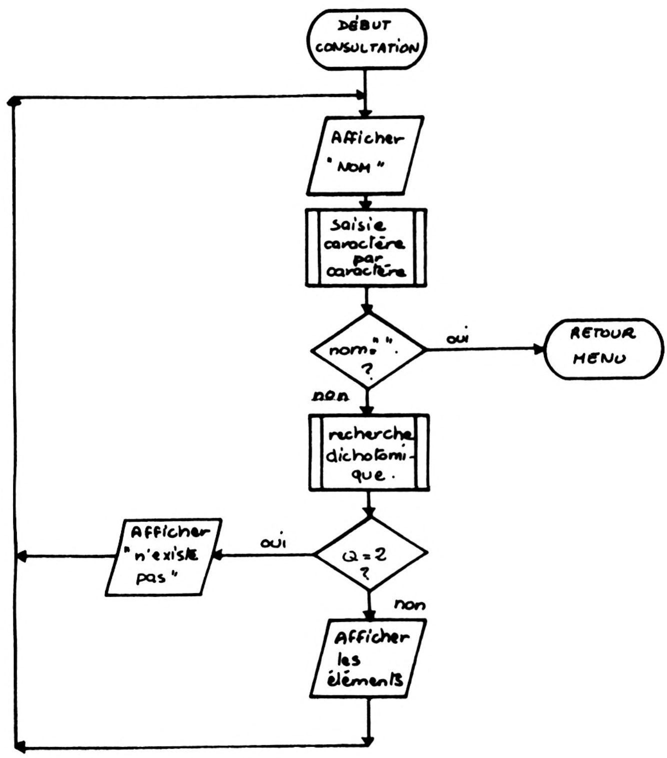
ORGANIGRAMME GÉNÉRAL

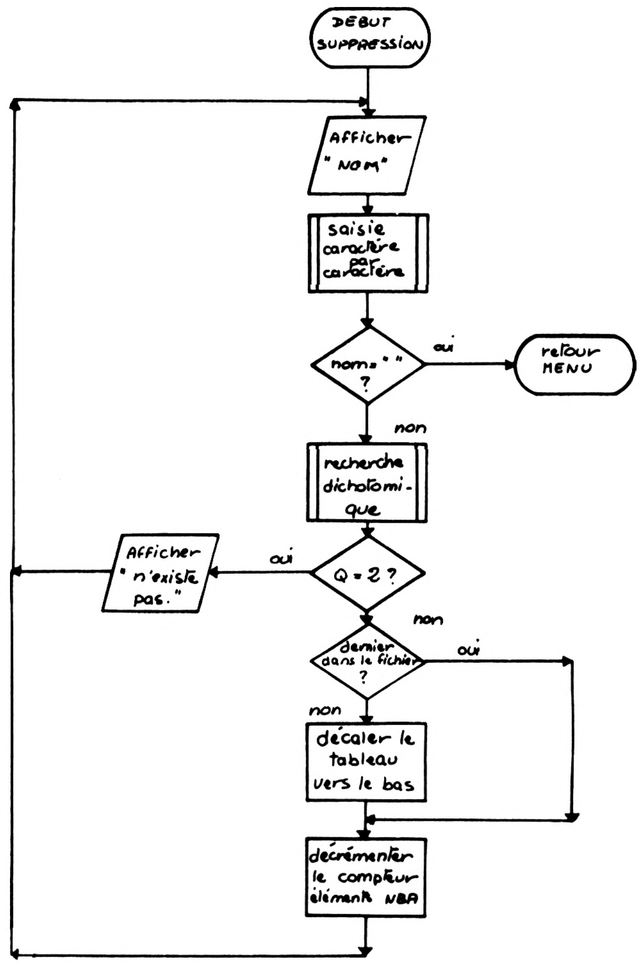


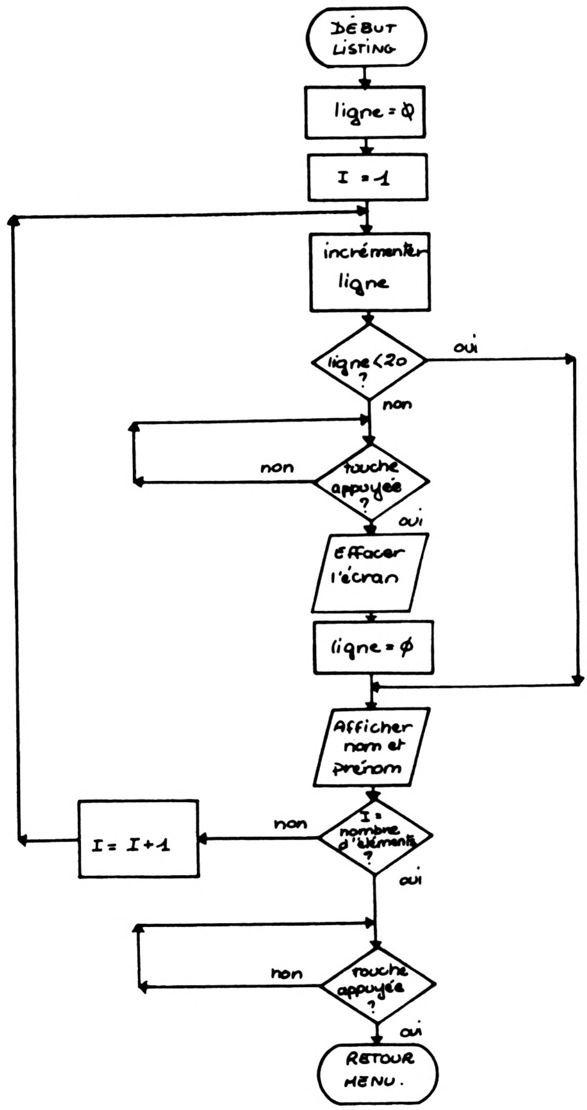

DEBUT sous. progr. recherche dichotom.
inférieur = 1
supérieur = nbre d'éléments
sup. < 1 ?
oui
non
inf. > sup. ?
oui
non
Inexistant Q = 2
milieu = (inf+sup.)/2
nom (milieu) = nom cherché ?
oui
non
trouvé Q = 1
retour au programme d'appel
nom (milieu) > nom cherché ?
oui
non
sup. = milieu - 1
inf. = milieu + 1


début saisie caractère par caractère
pointeur caractère = Ø (variable X)
variable tampon = " " (variable Y$)
saisie d'un caractère → X$
X$ = " " ?
oui
non
code X$ = new line ?
oui
non
retour programme d'appel
code X$ = Rubout
oui
non
X = 21 ?
oui
non
X = 1
oui
non
Y$(X) = X$ carac. saisi ds var. tampon
Afficher caractère
incrémenter pointeur X
Décrémenter pointeur X
Afficher un blanc à la position X
placer un blanc ds la variable tampon.


DÉBUT CREATION
Afficher "NOM"
Saisie caractère par caractère
Nom = " " ?
oui
RETOUR MENU
non
recherche dichotomique.
Q = 1 ?
oui
Afficher "EXISTE DÉJA"
non
Afficher "Je TRIE"
incrémenter compteur d'éléments
dernier dans le fichier ?
oui
non
Décaler le tableau vers le haut
placer le nom dans le tableau
Entrer prénom, adresse, code postal, ville.
placer ces éléments ds le tableau
utilisation du sous - programme de saisie caractère par caractère.


DÉBUT CONSULTATION
Afficher "NOM"
saisie caractère par caractère
nom=""?
oui
RETOUR MENU
non
recherche dichotomique.
Q=2 ?
oui
Afficher "n'existe pas"
non
Afficher les éléments


DEBUT
SUPPRESSION
Afficher
" NOM "
saisie
caractère
par
caractère
nom = " "
?
oui
retour
MENU
non
recherche
dichotomique
Q = 2 ?
oui
Afficher
" n'existe
pas. "
non
dernier
dans le fichier
?
oui
non
décaler le
tableau
vers le bas
décrémenter
le compteur
élément NBA


DEBUT LISTING
ligne = Ø
I = 1
incrémenter ligne
ligne<20 ?
oui
non
touche appuyée ?
non
oui
Effacer l'écran
ligne = Ø
Afficher nom et prénom
I = nombre d'éléments ?
non
I = I+1
oui
touche appuyée ?
non
oui
RETOUR MENU.

LE PROGRAMME

Lignes 1000 à 1140

Le programme utilise une recherche dichotomique qui sert aussi bien à la création, à la consultation et à la suppression.

Imaginons la recherche d'un mot dans un dictionnaire. On pourrait tout simplement le consulter de la première à la dernière page (METHODE SEQUENTIELLE). Mais cela risque d'être long, surtout si le mot commence par Z ! LA METHODE PAR DICHOTOMIE consiste à ouvrir le dictionnaire au milieu et à regarder si le mot trouvé est inférieur ou supérieur au mot recherché. Si le mot est inférieur, on prend la partie supérieure et on la partage en deux ; même chose si le mot est supérieur en ce qui concerne la partie inférieure. On repète le processus le nombre de fois nécessaire pour trouver le mot recherché.

Ici, la variable I représente le pointeur inférieur, et la variable S, le pointeur supérieur (nombre de noms déjà saisis). Si S est inférieur à 1 ou I supérieur à 5, le nom est inexistant dans le fichier. On saute alors directement à la ligne 1110 pour mettre Q=2, ce qui indique, au programme d'appel, que ce nom n'est pas dans le fichier.

On attribue, à la variable M, le milieu du tableau en divisant par 2 la somme de I et de S (ligne 1040). Si le nom pointé dans le tableau est identique au nom recherché, on va en ligne 1130 pour mettre Q=1, ce qui indique que le nom est bien dans le fichier.

Si l'article pointé est supérieur, on diminue le tableau par le haut (S=M-1).

Dans le cas contraire, on diminue le tableau par le bas (I=M+1).

Les autres lignes du programme sont facilement explicables en utilisant conjointement les organigrammes et les tableaux "identification des variables".

Exemple de recherche dichotomique sur un fichier de prénoms :

A( )

On recherche Suzanne

N°	PRÉNOM			
1	ALAIN	I = 1		
2	ARTHUR			
3	ERIC			
4	FRANÇOIS			
5	FREDERIC			
6	GEORGES			
7	HENRI			
8	ISABELLE	M = 8		
9	JULES		I = 9	
10	LILIANE			
11	ODILE			
12	PASCAL		M = 12	
13	RAYMOND			I = 13
14	SUZANNE			M = 14
15	THIERRY			
16	ZOÉ	S = 16	S = 16	S = 16

```
  10 REM GESTION DE FICHIERS
  20 REM ------------------
  30 DIM N$(100,10)
  40 DIM P$(100,10)
  50 DIM A$(100,20)
  60 DIM C$(100,5)
  70 DIM V$(100,20)
  80 DIM R$(10)
  90 LET NBA=0
 100 LET M=1
 110 DIM Y$(20)
 480 REM -------------------
 490 REM MENU
 500 CLS
 510 PRINT TAB 10;"*-= MENU =-*"
 520 PRINT
 530 PRINT
 540 PRINT TAB 8;"1  CREATION  1"
 550 PRINT
 560 PRINT TAB 7;"2  CONSULTATION  2"
 570 PRINT
 580 PRINT TAB 7;"3  SUPPRESSION  3"
 590 PRINT
 600 PRINT TAB 9;"4  LISTING    4"
 610 PRINT
 620 PRINT TAB 9;"5  EDITION    5"
 630 PRINT
 640 PRINT TAB 7;"6  SAUVEGARDE  6"
 650 PRINT
 660 PRINT
 670 PRINT
 680 PRINT TAB 8;"VOTRE CHOIX :"
 690 IF INKEY$="" THEN GOTO 690
 700 LET C=CODE INKEY$
 720 IF C<29 OR C>34 THEN GOTO 690
 730 LET C=C-27
 740 GOTO C*1000
 980 REM ---------------------
 990 REM RECHERCHE DICHOTOMIQUE
1000 LET I=1
1010 LET S=NBA
1020 IF S<1 THEN GOTO 1110
1030 IF I>S THEN GOTO 1110
1040 LET M=INT ((I+S)/2)
1050 IF N$(M)=R$ THEN GOTO 1130
1060 IF N$(M)>R$ THEN GOTO 1090
1070 LET I=M+1
1080 GOTO 1020
1090 LET S=M-1
1100 GOTO 1020
1110 LET Q=2
1120 RETURN
1130 LET Q=1
1140 RETURN
```

```
1180 REM -------------------
1190 REM SAISIE CARACTERE PAR
     CARACTERE
1200 LET X=1
1201 LET Y$=""
1205 LET X$=INKEY$
1210 IF X$="" THEN GOTO 1205
1220 IF CODE X$=118 THEN RETURN
1225 IF CODE X$=119 OR CODE X$=1
14 THEN GOTO 1300
1230 IF X=21 THEN GOTO 1205
1235 LET Y$(X TO X)=X$
1240 PRINT X$;
1250 LET X=X+1
1260 GOTO 1205
1300 IF X=1 THEN GOTO 1205
1310 LET X=X-1
1320 PRINT AT Y,10+X;" ";
1330 PRINT AT Y,10+X;
1340 LET Y$(X TO X)=" "
1350 GOTO 1205
1980 REM -------------------
1990 REM CREATION
2000 CLS
2010 PRINT TAB 10;"CREATION"
2020 PRINT AT 6,1;"NOM         ";
2030 LET Y=6
2040 GOSUB 1200
2050 IF CODE Y$=0 THEN GOTO 500
2060 LET R$=Y$
2100 GOSUB 1000
2110 IF Q=1 THEN GOTO 2700
2120 PRINT AT 21,10;"JE TRIE"
2130 IF NBA=0 THEN GOTO 2150
2140 IF N$(M)<R$ THEN LET M=M+1
2150 LET NBA=NBA+1
2160 IF M=NBA THEN GOTO 2225
2200 FOR J=NBA TO M+1 STEP -1
2210 LET N$(J)=N$(J-1)
2212 LET P$(J)=P$(J-1)
2214 LET A$(J)=A$(J-1)
2216 LET C$(J)=C$(J-1)
2218 LET V$(J)=V$(J-1)
2220 NEXT J
2225 PRINT AT 21,10;"       "
2230 LET N$(M)=R$
2240 PRINT AT 8,1;"PRENOM      ";
2250 LET Y=8
2260 GOSUB 1200
2270 LET P$(M)=Y$
2280 PRINT AT 10,1;"ADRESSE     ";
2290 LET Y=10
2300 GOSUB 1200
2310 LET A$(M)=Y$
2320 PRINT AT 12,1;"CODE P.     ";
2330 LET Y=12
2340 GOSUB 1200
2350 LET C$(M)=Y$
2360 PRINT AT 14,1;"VILLE       ";
2370 LET Y=14
2380 GOSUB 1200
2390 LET V$(M)=Y$
2400 GOTO 2000
2700 PRINT AT 21,4;R$;" EXISTE D
EJA"
2710 IF INKEY$="" THEN GOTO 2710
2720 GOTO 2000
```

```
2980 REM --------------------
2990 REM CONSULTATION
3000 CLS
3010 PRINT TAB 8;" CONSULTATION "
"
3020 PRINT AT 6,1;"NOM         ";
3030 LET Y=6
3035 GOSUB 1200
3036 IF CODE Y$=0 THEN GOTO 500
3040 LET R$=Y$
3045 GOSUB 1000
3050 IF O=2 THEN GOTO 3300
3060 PRINT
3070 PRINT AT 8,1;"PRENOM      ";P
$(M)
3080 PRINT
3090 PRINT AT 10,1;"ADRESSE    ";
A$(M)
3100 PRINT
3110 PRINT AT 12,1;"CODE P.    ";
C$(M)
3120 PRINT
3130 PRINT AT 14,1;"VILLE      ";
V$(M)
3140 PRINT AT 21,6;"TAPEZ UNE TO
UCHE"
3150 IF INKEY$="" THEN GOTO 3150
3160 GOTO 3000
3300 PRINT AT 21,0;Y$;"INEXISTAN
T"
3310 IF INKEY$="" THEN GOTO 3310
3320 GOTO 3000
3980 REM --------------------
3990 REM SUPPRESSION
4000 CLS
4010 PRINT TAB 10;" SUPPRESSION "
"
4020 PRINT AT 6,1;"NOM         ";
4030 LET Y=6
4040 GOSUB 1200
4050 IF CODE Y$=0 THEN GOTO 500
4060 LET R$=Y$
4070 GOSUB 1000
4080 IF O=2 THEN GOTO 4300
4090 IF M=NBA THEN GOTO 4170
4100 FOR I=M TO NBA-1
4110 LET N$(I)=N$(I+1)
4120 LET P$(I)=P$(I+1)
4130 LET A$(I)=A$(I+1)
4140 LET C$(I)=C$(I+1)
4150 LET V$(I)=V$(I+1)
4160 NEXT I
4170 LET NBA=NBA-1
4180 GOTO 4000
4300 PRINT AT 21,0;Y$;"INEXISTAN
T"
4310 IF INKEY$="" THEN GOTO 4310
4320 GOTO 4000
```

```
4980 REM --------------------
4990 REM LISTING
5000 CLS
5005 LET L=0
5010 FOR I=1 TO NBA
5020 LET L=L+1
5030 IF L<20 THEN GOTO 5050
5040 PRINT
5041 PRINT "TAPEZ UNE TOUCHE POU
R LA SUITE"
5042 IF INKEY$="" THEN GOTO 5042
5044 CLS
5046 LET L=0
5050 PRINT N$(I);" ";P$(I)
5060 NEXT I
5070 PRINT
5080 PRINT "TAPEZ UNE TOUCHE POU
R LE MENU"
5090 IF INKEY$="" THEN GOTO 5090
5100 GOTO 500
5980 REM --------------------
5990 REM EDITION
6000 CLS
6010 LPRINT TAB 5;"*-= LISTING F
ICHIER =-*"
5015 LPRINT
6020 FOR I=1 TO NBA
6030 LPRINT N$(I);" ";P$(I)
6040 LPRINT A$(I)
6050 LPRINT C$(I);" ";V$(I)
6060 LPRINT
6070 NEXT I
6080 GOTO 500
6980 REM ---------------------
6990 REM SAUVEGARDE
7000 CLS
7010 PRINT TAB 10;"SAUVEGARDE"
7020 PRINT
7030 PRINT " QUAND VOTRE MAGNETO
PHONE"
7040 PRINT " EST EN POSITION ENR
EGISTREMENT"
7050 PRINT " TAPER SUR UNE TOUCH
E POUR"
7060 PRINT " LA SAUVEGARDE DE VO
TRE FICHIER"
7070 IF INKEY$="" THEN GOTO 7070
7080 CLS
7090 FOR I=1 TO 16
7100 PRINT TAB I;"TAPEZ ""GOTO 5
00"""
7110 NEXT I
7120 PRINT
7130 PRINT "ET SURTOUT PAS ""RUN
"""
7140 SAVE "FICHIER"
```

DATE	NOM DU PROGRAMME : GESTION DE FICHIER	PAGE 1/2.

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A$( )	TABLEAU		X				Tableau des adresses
C	VARIABLE	X					choix de l'option (de 1 à 6)
C$( )	TABLEAU		X				Tableau des codes postaux
I	VARIABLE		X				Pointeur de la partie inférieure
							du tableau des noms dans la
							recherche dichotomique
J	INDICE		X				Boucle FOR / NEXT
L	COMPTEUR		X				Nombre de lignes affichées
							sur l'écran dans l'option listing.
M	VARIABLE		X				Milieu du tableau des noms
							dans la recherche dichotomique
NBA	COMPTEUR		X				Nombre de noms dans le
							fichier.
N$( )	TABLEAU		X				Tableau des noms
P$( )	TABLEAU		X				Tableau des prénoms
Q	AIGUILLAGE		X				Si Q = 1 le nom cherché existe
							Si Q = 2 le nom cherché n'existe pas
R$	VARIABLE		X				Nom à trouver dans le tableau
							dans la recherche dichotomique.
S	VARIABLE		X				Pointeur de la partie supérieure
							du tableau des noms dans
							la recherche dichotomique.

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

DATE	NOM DU PROGRAMME : GESTION DE FICHIER	PAGE 2/2

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
X	VARIABLE		X				Position du prochain caractère
							à afficher dans la saisie
							caractère par caractère
X$	VARIABLE	X					caractère saisi au clavier
Y	VARIABLE		X				Numéro de la ligne en cours
							dans la saisie caractère par
							caractère.
Y$	VARIABLE		X				Variable tampon de la saisie
							caractère par caractère.

6.1980

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Addition

Finis les ardoises et les cahiers d'essai pour le calcul mental ! Ce programme d'addition va permettre à vos enfants un entraînement efficace avec un correcteur patient qui ne se fâchera jamais.

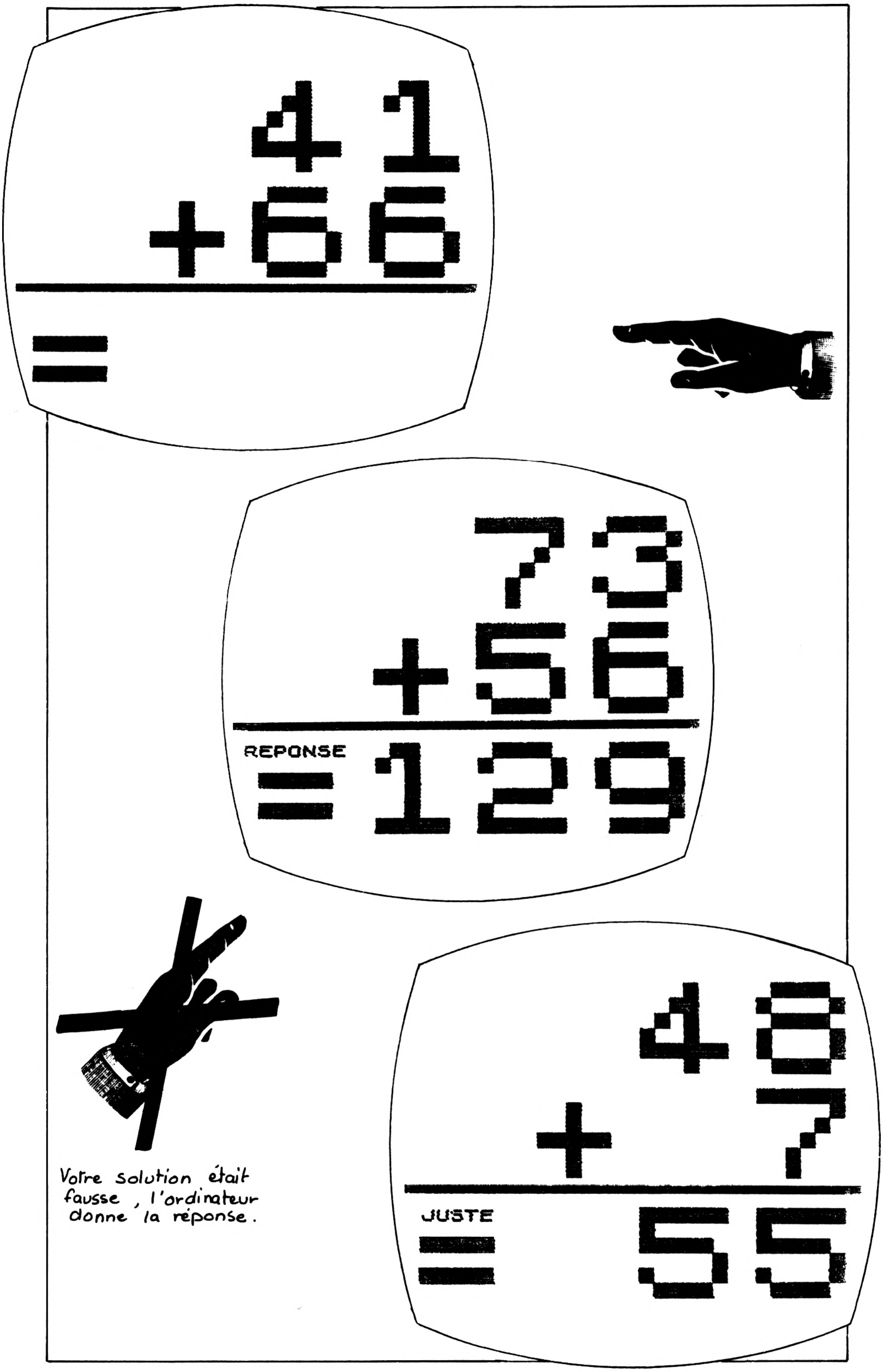
A la première utilisation du programme, vous taperez **RUN** ; ne vous inquiétez pas si l'écran se brouille un instant, c'est que le ZX81 charge un tableau des chiffres de 0 à 9 qu'il vous présentera tour à tour, ainsi que les signes + et =.

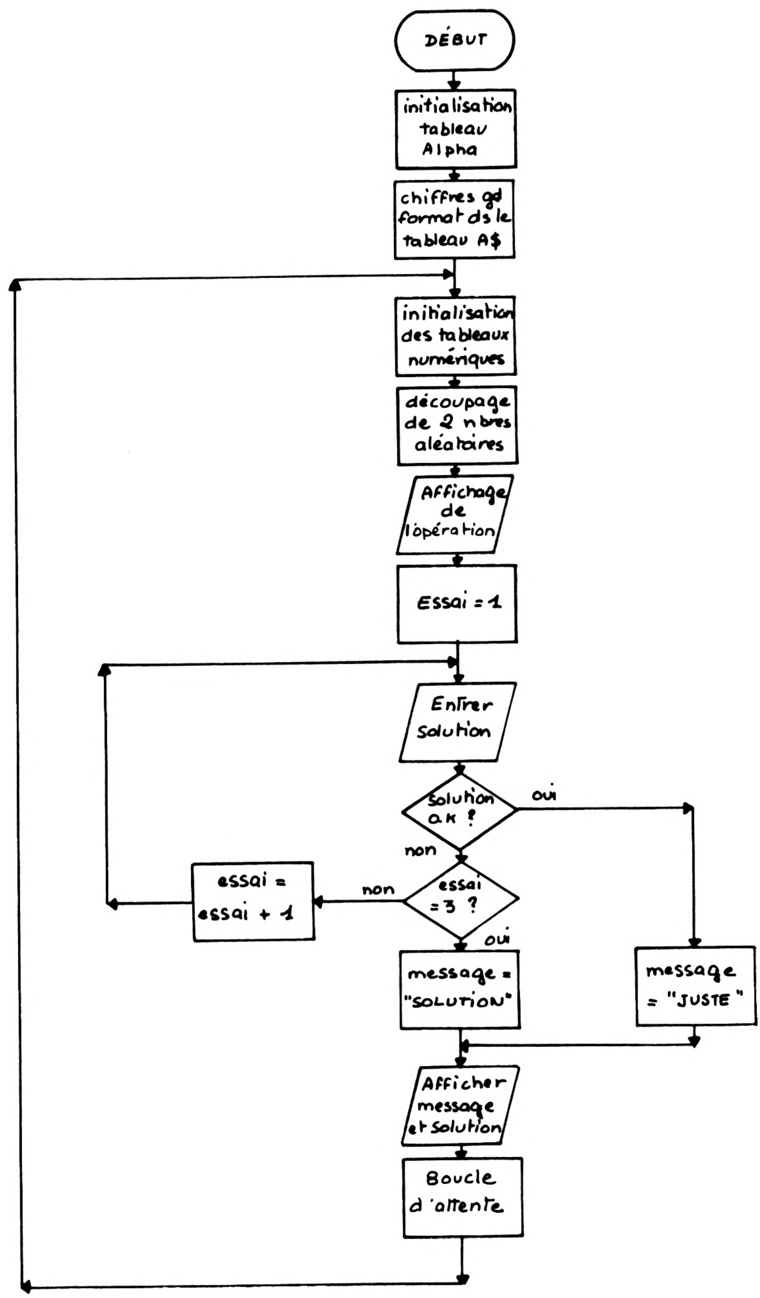
Après ce premier chargement, si vous interrompez le programme par **BREAK** ou si vous le réutilisez après l'avoir sauvegardé (lui et aussi son tableau, puisqu'au moment de la sauvegarde le ZX81 considère la mémoire entière), vous ne devez jamais taper **RUN**, mais GOTO 400, sinon vous n'auriez plus le contenu du tableau remis à zéro.

L'utilisation du programme est simple. Une addition en gros chiffres est proposée sur l'écran. Le ZX81 demande 3 fois la réponse en bas, à gauche de l'écran [L].

Si elles sont toutes fausses, il affichera de lui-même le bon résultat, puis reposera une nouvelle addition au bout de quelques instants.

Il fera de même si la bonne réponse est donnée en premier, deuxième ou troisième coup.


41
+66
=
73
+56
REPONSE
=129
Votre solution était fausse, l'ordinateur donne la réponse.
48
+ 7
JUSTE
= 55


DÉBUT
initialisation tableau Alpha
chiffres gd format ds le tableau A$
initialisation des tableaux numériques
découpage de 2 nbres aléatoires
Affichage de l'opération
Essai = 1
Entrer solution
Solution o.k ?
oui
non
essai = 3 ?
non
essai = essai + 1
oui
message = "SOLUTION"
message = "JUSTE"
Afficher message et solution
Boucle d'attente

```
  10 REM ADDITION
  20 REM --------------------
  50 DIM A$(13,8,8)
  60 DIM B$(8)
  70 LET Q=0
  90 FAST
 100 FOR I=0 TO 12
 105 IF I>9 THEN LET Q=-18
 110 FOR J=0 TO 6
 120 LET P=PEEK (7680+(8*(28+I+Q
))+J)
 130 FOR K=7 TO 0 STEP -1
 140 IF P<2**K THEN GOTO 170
 150 LET B$(8-K TO 8-K)="█"
 160 LET P=P-2**K
 170 NEXT K
 180 LET A$(I+1,J+1)=B$
 190 LET B$=""
 200 NEXT J
 210 CLS
 220 SLOW
 230 FOR K=1 TO 7
 240 PRINT A$(I+1,K)
 250 NEXT K
 260 FAST
 270 NEXT I
 380 REM -------------------
 390 REM LE JEU
 400 DIM D(2,2)
 410 DIM C(2)
 420 DIM E(3)
 430 SLOW
 440 CLS
 500 FOR I=1 TO 2
 510 LET C(I)=INT (RND*99)+1
 530 IF C(I)>9 THEN GOTO 560
 540 LET D(I,2)=C(I)
 550 GOTO 580
 560 LET D(I,1)=INT (C(I)/10)
 570 LET D(I,2)=C(I)-D(I,1)*10
 580 NEXT I
 600 FOR I=1 TO 2
 610 FOR J=1 TO 2
 620 IF D(I,J)=0 AND J=1 THEN GO
TO 660
 630 FOR K=1 TO 7
 640 PRINT AT ((I-1)*7)+K-1,8+(J
*8);A$(D(I,J)+1,K)
 650 NEXT K
 660 NEXT J
 670 NEXT I
 680 FOR I=0 TO 31
 690 PRINT AT 14,I;"▄"
 695 NEXT I
 700 FOR I=11 TO 12
 710 FOR K=1 TO 7
 720 PRINT AT ((13-I)*8)+K-2,(I-
11)*8;A$(I,K)
 730 NEXT K
 740 NEXT I
 800 LET T=C(1)+C(2)
 810 FOR I=1 TO 3
 820 INPUT S
 830 IF S=T THEN GOTO 870
 840 NEXT I
 850 LET S$="REPONSE"
 860 GOTO 900
 870 LET S$="  JUSTE"
 900 PRINT AT 16,1;S$
 910 IF T>100 THEN GOTO 980
 920 IF T>10 THEN GOTO 950
 930 LET E(3)=T
 940 GOTO 1100
 950 LET E(2)=INT (T/10)
 960 LET E(3)=T-E(2)*10
 970 GOTO 1100
 980 LET E(1)=1
 990 LET E(2)=INT ((T-100)/10)
1000 LET E(3)=T-100-E(2)*10
1100 FOR I=1 TO 3
1105 IF E(I)=0 AND I=1 THEN GOTO
 1140
1110 FOR K=1 TO 7
1120 PRINT AT 14+K,(I*8);A$(E(I)
+1,K)
1130 NEXT K
1140 NEXT I
1200 FOR I=1 TO 80
1210 NEXT I
1220 GOTO 400
```

LE PROGRAMME

Lignes 50 à 220

On dimensionne un tableau A$ pour y placer les chiffres en grand format. On passe en mode rapide à la ligne 90 pour gagner du temps sur la constitution de ce tableau. On va chercher, dans la mémoire morte du ZX81, la matrice de tous les chiffres que l'on décode et que l'on place dans le tableau (voir les explications dans le programme MACRO-LETTRES). On place chaque point de chaque ligne du chiffre traité dans la variable tampon B$. Quand un chiffre est constitué, on l'affiche sur l'écran (pour faire patienter l'utilisateur) en passant du mode rapide au mode lent.

Lignes 400 à 800

On place, dans le tableau C(), deux nombres aléatoires compris entre 1 et 99 pour constituer l'addition. On décompose chaque nombre dans le tableau D() pour le cadrage sur l'écran des chiffres en grand format ; puis, on les affiche un à un, ainsi que les signes + et = et le trait du total.

Lignes 810 à 900

On demande trois fois la solution par l'intermédiaire de la **boucle FOR/NEXT.** Si une réponse est bonne, on passe directement à la ligne 850 pour placer, dans la variable S$, le message "JUSTE".

Si aucune réponse n'est juste, on place, dans S$, le message "REPONSE". La ligne 900 l'affiche.

Lignes 910 à 1220

Quelles que soient les réponses, bonnes ou mauvaises, le programme affiche la solution.

Pour cela, on utilise la même méthode de découpage de nombres en chiffres, mais en les plaçant cette fois dans le tableau E(). Le découpage est un peu plus complexe puisque la solution peut comporter trois chiffres.

Après une boucle d'attente à la ligne 1200, on revient en 400 pour une nouvelle addition.

DATE	NOM DU PROGRAMME : ADDITION	PAGE 1

IDENTIFICATION des VARIABLES

NOM SYMBOLIQUE	TYPE (*)	ORIGINE Clavier	ORIGINE Programme	ORIGINE Cassette	ORIGINE Disquette	LONG.	OBSERVATIONS
A$(,)	TABLEAU		X				Tableau contenant les chiffres
							grand format.
B$	VARIABLE		X				Variable tampon pour l'élaboration
							des gros caractères
C(,)	TABLEAU		X				Contient les nombres aléatoires
							de l'addition.
D(,)	TABLEAU		X				Contient le découpage des chiffres
							à afficher.
E(,)	TABLEAU		X				Contient le découpage des chiffres
							de la solution.
I	INDICE		X				Boucle FOR / NEXT
J	INDICE		X				Boucle FOR / NEXT
K	INDICE		X				Boucle FOR / NEXT
P	VARIABLE		X				Matrice du caractère à imprimer
Q	VARIABLE		X				Aiguille la matrice de caractères
							des chiffres aux signes + et =
S	VARIABLE	X					Solution proposée
S$	VARIABLE		X				Solution juste ou fausse
T	VARIABLE		X				Total de l'addition.

(*) Légende : aiguillage, compteur, constante, index, indice, tableau, variable.

Achevé d'imprimer en juin 1982
sur les presses de l'imprimerie Laballery et Cie
58500 Clamecy
Dépôt légal : juin 1982

n° d'impression : 20601
N° d'édition : 86595-42-5
ISBN : 2-86595-042-5

PROGRAMMES**

JEAN-FRANÇOIS SEHAN

ETUDES POUR ZX 81

Cet ouvrage, qui s'adresse aux possesseurs de ZX 81, constitue un recueil de programmes BASIC. Ceux-ci utilisent au mieux les possibilités de graphisme et de création de fichiers sur cassette. Du jack-pot à la guerre des étoiles, en passant par la fusée lunaire et la gestion de fichier, ce livre regroupe vingt programmes les plus divers.

Editions du P.S.I.

Boîte Postale 86

F-77400 Lagny/Marne

ISBN 2-86595-042-5 — Imprimé en France

Couverture Graphic And Co

P.S.I. PROGRAMMES

ETUDES POUR ZX 81

JEAN-FRANÇOIS SEHAN

JEAN-FRANÇOIS
SEHAN

ETUDES POUR ZX 81

20 PROGRAMMES EN BASIC

P.S.I.

EDITIONS DU P.S.I.

PROGRAMMES

PROGRAMMES**

JEAN-FRANÇOIS SEHAN

ETUDES POUR ZX 81

Cet ouvrage, qui s'adresse aux possesseurs de ZX 81, constitue un recueil de programmes BASIC. Ceux-ci utilisent au mieux les possibilités de graphisme et de création de fichiers sur cassette. Du jack-pot à la guerre des étoiles, en passant par la fusée lunaire et la gestion de fichier, ce livre regroupe vingt programmes les plus divers.

Editions du P.S.I.
Boîte Postale 86
F-77400 Lagny/Marne

ISBN 2-86595-042-5 — Imprimé en France

Couverture Graphic And Co

P.S.I. PROGRAMMES

ETUDES POUR ZX 81

JEAN-FRANÇOIS SEHAN

AMSTRAD CPC

MÉMOIRE ÉCRITE
MEMORY ENGRAVED
MEMORIA ESCRITA

https://acpc.me/